DOCUMENT RESUME

ED 304 589 CE 052 133

AUTHOR Mills, Steven; And Others
TITLE Welcome to dBase III Plus. Learning Activity Packets.
INSTITUTION Oklahoma State Dept. of Vocational and Technical Education, Stillwater. Curriculum and Instructional Materials Center.
PUB DATE 89
NOTE 148p.; For a related document, see CE 052 134.
AVAILABLE FROM Curriculum and Instructional Materials Center, Oklahoma Department of Vocational and Technical Education, 1500 West Seventh Avenue, Stillwater, OK 74074-4364 (Order No. BO-3021).
PUB TYPE Guides - Classroom Use - Guides (For Teachers) (052)

EDRS PRICE MF01/PC06 Plus Postage.
DESCRIPTORS *Business Education; Computer Assisted Instruction; Computer Software; Database Design; *Database Management Systems; *Databases; Data Processing; *Individualized Instruction; Learning Activities; Learning Modules; Management Systems; Office Occupations Education; Pacing; Postsecondary Education; Secondary Education
IDENTIFIERS *dBase III Plus

ABSTRACT

This learning activity packet (LAP) contains nine self-paced study lessons that allow students to proceed along a a 43-hour course of study for dBase III Plus at their own pace. The lessons are organized in the following way: objectives, completion standard, performance standard, a list of required materials, unit test, and exercises (applications of the operations to measure the laboratory competency of the student in regard to the materials presented in the LAP). Most lessons are presented in a "read and do" format. The lessons are: database concepts; using the dBase III Plus Assistant; creating a database; adding data to a database file; modifying data in a database file; report and mailing label preparation; memory variables, operations, and functions; sorting and indexing files; and data entry forms. Unit tests for each lesson and answer keys for the tests and exercises are appended. These lessons have been prepared for use with the IBM Personal Computer. (YLB)

****
* Reproductions supplied by EDRS are the best that can be made *
* from the original document. *
****

LEARNING ACTIVITY PACKETS

ED 304 589

U.S. DEPARTMENT OF EDUCATION
Office of Educational Research and Improvement
EDUCATIONAL RESOURCES INFORMATION CENTER (ERIC)

☑ This document has been reproduced as received from the person or organization originating it.
☐ Minor changes have been made to improve reproduction quality.

• Points of view or opinions stated in this document do not necessarily represent official OERI position or policy.

"PERMISSION TO REPRODUCE THIS MATERIAL HAS BEEN GRANTED BY

R Matkin

TO THE EDUCATIONAL RESOURCES INFORMATION CENTER (ERIC)."

Developed by
Curriculum and Instructional Materials Center
Oklahoma State Board of Vocational and Technical Education

CE052133

89-006115

80 3021

LEARNING ACTIVITY PACKETS

Developed by
Curriculum and Instructional Materials Center
Oklahoma State Board of Vocational and Technical Education

Copyright 1988
State Department of Vocational and Technical Education
Curriculum and Instructional Materials Center

All rights reserved.

Printed in the United States of America by the
State Department of Vocational and Technical Education
Stillwater, Oklahoma 74074

This publication, or parts thereof, may not be reproduced in any form by photographic, electrostatic, mechanical, or any other method, for any use including information storage and retrieval, without written permission from the publisher.

Use of commercial products in these instructional materials does not imply endorsement by the Oklahoma State Department of Vocational and Technical Education.

The Oklahoma State Department of Vocational and Technical Education does not discriminate on the basis of race, creed, color, national origin, sex, age, veteran status, or qualified handicap.

88-006115

ACKNOWLEDGEMENTS

The development of the *dBase III Plus© Learning Activities Packet* is credited to Steven Mills and Gayle Kearns. It was their hard work and effort that made the LAPs a success.

A great appreciation is extended to Sheila Stone, Curriculum Coordinator for her coordination during the development of the Learning Activity Packets. A special thanks goes to Terrell Ogilvie, Technical Writer, for his contribution toward the development of the LAPs.

A special thanks goes to the Business and Office instructors who participated in the training workshops and helped validate and refine the LAPs.

PREFACE

WHAT ARE LAPS?

Learning Activity Packets (LAPs) are self-paced study lessons that allow the student to proceed along a course of study at his or her own pace. The student is allowed to progress to the next lesson in the unit only after a certain level of competency and understanding is achieved.

The advantage to the student of using LAPs for certain areas of study is obvious in that it allows the student to progress at his or her own pace. Some students learn faster than others and LAPs takes this fact into consideration by allowing all to learn at their own rate.

The advantage to the instructor of using LAPs for certain areas of study is that the instructor becomes available to be your lab partner. This allows more individual attention to each student's learning needs.

ORGANIZATION OF LESSONS

The lessons used in the LAPs are organized in the following way:

LESSON NAME	This is the name of the LAP within the instructional unit.
LESSON NO.	Each lesson is numbered to maintain the right sequence of instruction
OBJECTIVES	A description of the major points or concepts which will be taught in a lesson as well as a description of what the student should be able to perform upon completion of the unit.
COMPLETION STANDARD	This is a suggested maximum amount of time the student will need in order to complete the lesson. This information is available in the Teachers Manual and Index.
PERFORMANCE STANDARD	This score must be achieved on the Unit Test and Exercises before progressing to the next unit. These LAPs will always use a standard of 70% for each lesson.
REQUIRED MATERIALS	This is a list of all materials the student will need in order to complete the lesson.

UNIT TEST — The Unit Test is used to measure the theory competency of the student in regard to the material presented in the LAP.

EXERCISES — An application of the operations is assigned to measure the laboratory competency of the student in regard to the material presented in the LAP.

READ AND DO

Most lessons are presented in a READ and DO format. Some information is presented in a READ paragraph. A response to the information just presented is made in the form of a DO paragraph. However, sometimes information is presented with no requisite DO responses.

COMPUTER HARDWARE CONFIGURATION

These lessons have been prepared for use with the IBM Personal Computer. Since the computer you use may be equipped differently from other IBM-PC's, you should take note as to the configuration of your computer. Many instructions may refer to a particular disk configuration or a particular kind of printer. You should be aware of the particular configuration for the micrcomputer your are using.

SOFTWARE VERSION

The information presented in the lessons generally reflect the latest version of the software currently available on the consumer market.

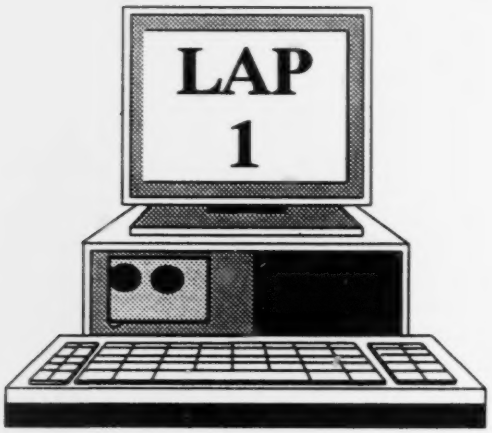
LEARNING PROCEDURES

This is the step-by-step procedure the student should follow in order to complete the lesson properly. Generally, the learning procedure for each lesson should follow this schedule:

1. READ the Objectives for the lesson. Be certain that you have selected the correct lesson and have the required materials.
2. READ and DO the information and operations presented in the lesson.
3. DO the Exercises at the end of the lesson and return to the instructor for scoring.
4. STUDY the information presented in the lesson.
5. DO the Unit Test and return to the instructor for scoring.
6. If you meet the performance standard, MOVE to the next lesson in the sequence.

dBASE III PLUS -- BEGINNING

NO.	LESSON NAME	PRE REQ.	COMPLETE. STANDARD
01	Database Concepts	None	2 hrs.
02	Using the dBASE III PLUS Assistant	01	2 hrs.
03	Creating a Database	02	3 hrs.
04	Adding Data to a Database File	03	6 hrs.
05	Modifying Data in a Database File	04	3 hrs.
06	Report and Mailing Label Preparation	05	6 hrs.
07	Memory Variables, Operations, and Functions	06	6 hrs.
08	Sorting and Indexing Files	07	9 hrs.
09	Data Entry Forms	08	6 hrs.
	TOTAL TRAINING TIME FOR COURSE		43 hrs.


LAP
1

LESSON 1

DATABASE CONCEPTS

OBJECTIVES:

* Define a database management system (DBMS)
* Describe the organization of a relational DBMS
* Identify dBASE III PLUS technical specifications
* Identify dBASE III PLUS file types
* Identify dBASE III PLUS field types

REQUIRED MATERIALS:

* Paper and pen or pencil

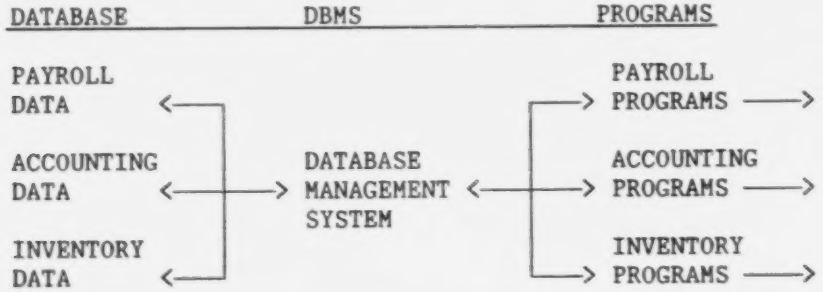
READ: DATABASE MANAGEMENT SYSTEMS VS. FILE PROCESSING SYSTEMS

A database management system (DBMS) like dBASE III PLUS is considerably different from a file processing system. A file processing system is usually configured like this:

FILES		PROGRAMS		OUTPUT
PAYROLL FILES	<———>	PAYROLL PROGRAMS	<———>	PAYROLL OUTPUT
ACCOUNTING FILES	<———>	ACCOUNTING PROGRAMS	<———>	ACCOUNTING OUTPUT
INVENTORY FILES	<———>	INVENTORY PROGRAMS	<———>	INVENTORY OUTPUT

The payroll programs process the payroll files. The accounting programs process the accounting files. And the inventory programs process the inventory files. To get reports that combine data from different files, a new program would have to be written and it wouldn't necessarily work: the data may be incompatible from file to file, or may be buried so deeply within the other programs that getting it out is more trouble than it's worth.

A Database Management System (DBMS) integrates the data and makes it easier to get useful information from your records, rather than just reams of data. Conceptually, a DBMS looks something like this:



Data is monitored and manipulated by the DBMS, not the individual applications programs. All of the applications systems have access to all of the data. In a file processing system, this would not require a great deal of duplicated data. Aside from the potential for entry errors, data integrity is extremely hard to maintain when the same data is supposed to be duplicated in different files--it never is.

To generate a new application in a file processing system, a new program and new files must be set up. Using a DBMS, a new access program is written, but the data does not have to be restructured--the DBMS takes care of it.

If a new kind of data is added to a record (salary history in a personnel file, for example), file processing programs have to be modified. With a DBMS, additions and changes have no effect on the programs that don't need to use the new information--they don't see it and don't know that it's there.

READ: RELATIONAL DATABASE MANAGEMENT SYSTEMS

Database Management Systems come in two flavors: hierarchical and relational. These terms refer to how the DBMS keeps track of data.

A hierarchical system tends to get extremely complex and difficult to maintain because the relationships between the data elements are maintained with sets, linked lists, and pointers telling the system where to go next. You can easily end up with lists of lists of lists and pointers to pointers to pointers.

A relational database management system is a great deal simpler. Data is represented as it is, and the relation between data elements can be considered a two-dimensional table like this one:

Column 1	Column 2	Column 3	Column 4	Column 5
Invoice #	Supplier	Description	Amount	Number
2386	Graphics Process	Prints	23.00	BBQ-747
78622	Brown Engraving	Litho plates	397.42	TFS-901
M1883	Air Freight, Inc.	Shipping	97.00	SPT-233

Each row going across the table is called a record. Each column is called a field of the record. Each entry in the table must be a single value (no arrays, no sets, etc.) All the entries in a column must be of the same type. Each record (row) is unique, and the order of records (rows) doesn't matter.

dBASE III PLUS adheres to the relational model for database management systems. This table consists of rows and columns. Each record in the table is assigned a record number:

		ROOM	GRADE	NAME
	001	11	2	Abbott, Anthony A.
Record	002	15	1	Adams, John Q.
Number ⟶	003	9	3	Allen, Steven C.
	. . .	. . .	. . .	. . . . .
	317	15	2	Zieger, Lenny P.

A relational database management system provides the capability to cross-reference between relations (database files in dBASE III PLUS). The figure below provides an illustration of the relational concept. Students and teachers can be related based upon the contents of the room number column (field). Anthony Abbott's teacher (Bobby Baker) is found by cross-referencing the two tables on the room number:

	ROOM	GRADE	STUDENT
001	11	2	Abbott, Anthony A.
002	15	1	Adams, John Q.
003	9	3	Allen, Steven C.
004	7		
005	11		
006	14		
007	18		

	ROOM	GRADE	TEACHER
001	14	6	Abernathy, Allen
002	11	2	Baker, Bobby
003	7	5	Clemson, Clementine
004	15	1	Farhquar, Francis

When we show you more realistic examples later, you'll see that relations don't get any more complicated, just larger.

READ: dBASE III PLUS TECHNICAL SPECIFICATIONS

DATABASE FILE	Number of records	-- 1 billion maximum
	Number of bytes	-- 2 billion maximum
	Record size	-- 4000 bytes in .dbf file
	Fields	-- 128 maximum
FIELD SIZE	Character fields	-- 254 bytes maximum
	Date fields	-- 8 bytes
	Logical	-- 1 byte maximum.
	Memo fields	-- 5000 bytes maximum or the capacity of word processor used
	Numeric fields	- 19 bytes maximum.
FILE OPERATIONS	15 open files of all types 10 open database files. A database file counts as two files if memo fields are used. 7 open index files per active database file. 1 open format file per active database file.	
NUMERIC ACCURACY	15.9 digits. Note that the decimal point does not count as a digit in determining accuracy. Largest number: $1 \times 10^{+99}$. Smallest positive number: 1×10^{-307}.	
MEMORY VARIABLES	Number of active memory variables: 256 Total bytes for memory variables: 6000 All values may be limited by the computer hardware configuration.	
COMMAND LINE	Maximum length	-- 254 characters

READ: DATABASE FILES

dBASE III PLUS saves information as disk files in thirteen specialized formats. Each serves a specific dBASE III PLUS processing need. Each disk file has a file name of not more than eight characters and a file extension consisting of a period and three characters.

A file name is assigned by the user at the time the file is created. The file extension must be assigned at the same time. Normally, dBASE III PLUS assigns the file extension according to the type of file being created. But, users can override the dBASE III PLUS assignment by simply typing in an extension of their own choosing at the time the file name is entered. The extension assigned by dBASE III PLUS is the default extension.

These extensions as well as the thirteen kinds of files, are defined in the following table:

KIND OF FILE	FILE TYPE
Catalog	.cat
Database	.dbf
Database Memo	.dbt
Index	.ndx
Memory Files	.mem
Command and Procedure	.prg
Format	.fmt
Label	.lbl
Report Form	.frm
Query	.qry
Screen	.scr
View	.vue
Text Output	.txt

If an extension other than the default extension is used, it must be included with the file name whenever the file is selected for use.

READ: CATALOG (.cat) FILES

Catalog files contain sets of related data resources (such as database files) and operational resources (such as .fmt, .frm, and .lbl files). They are created with the SET CATALOG TO command.

READ: DATABASE (.dbf) FILES

Database files store data records and fields (rows and columns). Each record is intended to contain a set of unique information.

dBASE III PLUS database files can hold up to 1 billion records. Each record can contain up to 4,000 bytes, which must be contained within no more than 128 data fields.

READ: DATABASE (.dbt) MEMO FILES

Database memo files are auxiliary files to a database (.dbf) file. They are used to store the contents of memo fields. All memo fields in a .dbf file are stored in the same .dbt file.

Each database record may contain up to 128 memo fields. Memo fields may contain the same kind of information as character fields, but can have as many as 5,000 characters. Each memo field requires a minimum of 512 bytes per record if any data is specified. Each memo field occupies 10 spaces in the database file, because the text is stored in a separate file on your disk.

READ: INDEX (.ndx) FILES

Index files provide the means to use a database in a logical order rather than the physical order. The physical order is the order in which the records were entered. Logical order is an alphabetical, chronological, or numerical order based on the contents of one or more database fields.

Index files relate a key (an item of interest, such as a name) to a corresponding database record number. When using a database with an index file, the database appears to be in the order of the key item.

The key, which can be one or more fields, may also be used to provide direct access to a particular record.

READ: MEMORY (.mem) FILES

Memory files contain up to 256 memory variables. They are used to save the contents of memory variables for later use. These files are created by the command SAVE and are read back into memory by the command RESTORE.

READ: COMMAND AND PROCEDURE (.prg) FILES

Command files contain sets of dBASE III PLUS instructions that have been stored as programs. They are ASCII files and may be created with either MODIFY COMMAND or the non-document mode of most word processing programs.

READ: FORMAT (.fmt) FILES

Format files create custom screen forms for use in data entry and printed output. They can be generated from .scr files set up by the CREATE SCREEN command or input directly using MODIFY COMMAND. Any word processor that creates ASCII text files may be used as well.

READ: LABEL (.lbl) FORM FILES

Label files contain the information needed by the LABEL command to print labels. They are created and altered by the CREATE/MODIFY LABEL command.

READ: REPORT FORM (.frm) FILES

Report form files contain the information needed by the REPORT command to prepare reports. They can be created and altered by the CREATE/MODIFY REPORT command.

READ: QUERY (.qry) FILES

Query files contain a filter condition that restricts the records that are displayed in commands that require a database (.dbf) file to be in USE. They are created with the CREATE/MODIFY QUERY command.

READ: SCREEN (.scr) FILES

Screen files contain information that is used by the CREATE/MODIFY SCREEN command to generate and format files.

READ: VIEW (.vue) FILES

View files contain the names of database files, their associated indexes, format files, a list of selected field names, filter conditions and relationships between these files which have been created with CREATE/MODIFY VIEW.

READ: TEXT OUTPUT (.txt) FILES

Text output files are primarily used as the means to interface dBASE III PLUS to other purchased software. These files are ASCII and contain only the printable ASCII characters.

Text output files will be read into a dBASE III PLUS database file by a special form of the APPEND FROM command. They are created by a special form of the COPY command. They can also be used to record dBASE III PLUS processing activity by means of the SET ALTERNATE command.

READ: DATABASE STRUCTURE AND FIELD TYPES

The structure or framework of a database file is established by defining each of the fields in the database. This is accomplished by means of the CREATE and MODIFY STRUCTURE commands.

Field definitions consist of:

FIELD NAMES — Field names may be up to 10 characters long. They must begin with a letter and cannot contain embedded blank spaces. Letters, numbers, and underscores are permitted.

FIELD WIDTH — Field width is the maximum number of characters or digits that are to be contained in the field. In the case of numeric fields, the decimal point (if any) is counted as a digit.

FIELD TYPES — The field types used in dBASE III PLUS are:

C - Character
D - Date
L - Logical
N - Numeric
M - Memo

READ: CHARACTER FIELDS

Character fields may be used to store any printable character that can be entered from the keyboard (including blank spaces). These are the printable ASCII characters. They include letters, numbers, special symbols, and blank spaces. The maximum field width (size) of a character field is 254.

READ: DATE FIELDS

Date fields are used to store dates. Dates may be entered and displayed in a variety of forms by using the PICTURE function. The default date entry and display format is mm/dd/yy. The field width of a date field is always 8.

A date may be added to a number; a date may be subtracted from a date, and a number may be subtracted from a date.

READ: NUMERIC FIELDS

Numeric fields are two types: integer and decimal. An integer is a number that does not have decimal places (for example, the number of students in a class). The field width of a numeric field is the number of digits the field can hold (the decimal point counts as one digit).

Accuracy of numeric fields extends to 15.9 digits (the decimal point does not count here). This means the most significant 15 digits of any number will be reliable.

READ: LOGICAL FIELDS

Logical fields accept only single characters representing True/False values. Logical True (.T.) is entered as T, t, Y, or y. Logical False (.F.) is entered as F, f, N, or n.

READ: MEMO FIELDS

Memo fields are designed to accommodate large blocks of textual information. They are stored in an auxiliary file to the database file. They are indicated in the .dbf file by the word "Memo."

The memo field size is variable. If no data is entered, the field size is 0. As data in entered field space is assigned in 512-byte blocks to a maximum field size of 4096 bytes. Each memo field uses 10 bytes in each database (.dbf) file record.

EXERCISES

Name:_____ Date:_____

(1) Build a database in the space below for the following data:

445-24-6789	588-89-9223	889-34-0182
Bill Bagley	Carol Carter	Dave Defore
Hired: 5/18/86	Hired: 4/22/86	Hired 8/13/85
Salesman	Engineer	Clerk
Marketing	Research	Accounting
22,000	25,000	18,000

Write a name for the file in the space provided below. Decide what fields to include and write the field names in the spaces provided below. Then enter the data for the three records into the spaces provided for field information:

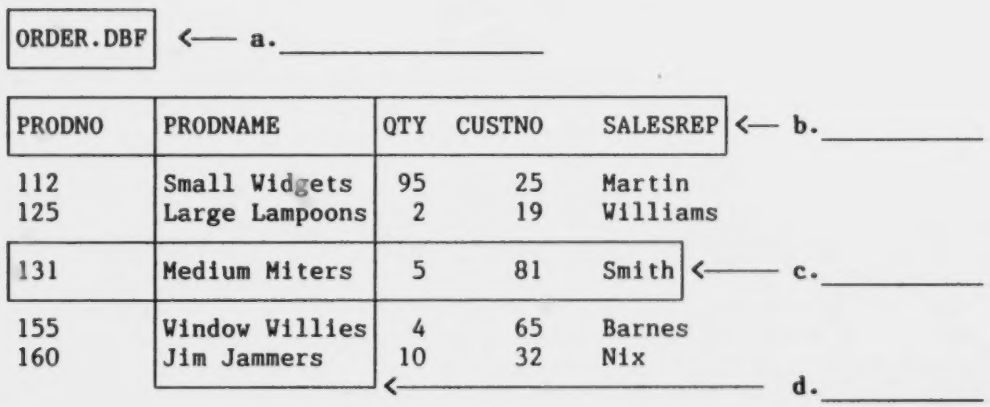
_____

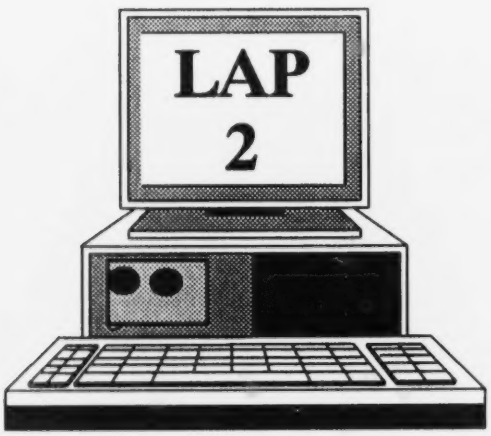
FILENAME

FIELD NAMES:						
1.						
2.						
3.						

(2) Identify the following parts of the database by writing the term or phrase from the list below on the appropriate line of the diagram:

Field Record Field Name Database File Name




LAP
2

LESSON 2

USING THE DBASE III PLUS ASSISTANT

OBJECTIVES:

* Describe the dBASE III PLUS working environment
* Perform the dBASE III PLUS power-on procedure
* Describe the dBASE III PLUS Assistant menus and screens
* Select the default drive
* Describe the HELP program
* Perform the dBASE III PLUS power-off procedure

REQUIRED MATERIALS:

* Paper and pen or pencil
* dBASE III PLUS System Diskettes 1 and 2
* A practice diskette
* IBM PC/XT, PS/II, or compatible with 2 disk drives or 1 disk drive and hard disk, and printer

READ: THE DBASE III PLUS WORKING ENVIRONMENT

dBASE III PLUS requires users to work interactively with the program by entering commands at a dot prompt. As a command-driven program, it consequently requires users to remember a great number of commands. Users keep command reference cards or manuals for ready reference. However, dBASE III PLUS provides some interactive help facilities to greatly improve the working environment.

READ: THE DBASE III PLUS ASSISTANT PROGRAM

The Assistant provides an alternate method for working with dBASE III PLUS. The Assistant provides a series of menus and other information to help you build your database files and generate reports. When using the Assistant program, each time you select a menu option, the next choice is presented until an entire dBASE III PLUS command is built.

From the beginning, the Assistant structures your interaction with dBASE III PLUS. As you proceed you see the dBASE commands that you have built without having to learn the extensive dBASE III PLUS command language. First of all, you will get acquainted with the menus and screens of the Assistant.

DO: DBASE III PLUS POWER-ON PROCEDURE

(1) Insert a DOS diskette in Drive A and your practice diskette in Drive B of the computer and power on the computer, **OR**, if you are using a computer with a hard disk drive, just power on the computer (insert your practice diskette in Drive A after the computer is powered on and the sign-on screen is displayed).

(2) At the DOS prompt, A>, (or C> if you are using a hard disk drive) is displayed, remove the DOS disk and insert the dBASE III PLUS System Disk 1 in Drive A.

Type: **dbase** **<Return>**

(3) dBASE III PLUS directs you to insert System Disk 2 in Drive A. So insert System Disk 2 in Drive A and your practice disk in Drive B and press <Return>.

OR

If you are using a hard disk system with a disk management or main menu program, just select the option on the menu for dBASE III PLUS.

READ: THE ASSIST SCREEN AND MENU BAR

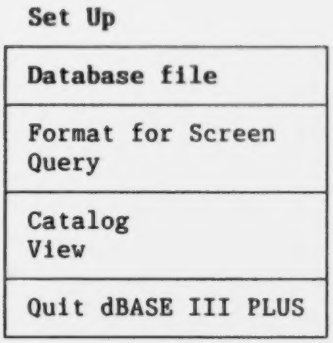
The Assistant program is now displayed on the screen. Notice the top line of the Assist screen with the words:

Set Up Create Update Position Retrieve Organize Modify Tools

This line is called the **menu bar**. It displays the main categories of commands, arranged in usual order of operation. The clock is blinking at the far right on this line. The highlighted words, **Set Up**, mean the selection bar is positioned on this option.

READ: PULL-DOWN MENUS

Notice underneath the words, **Set Up**, there is a vertical box with the following options:



This is a pull-down menu. All these functions are available options in the Set Up menu. The highlighted words, **Database file**, mean the selection bar is positioned on this option.

READ: STATUS BAR

Near the bottom of the screen is a line with the following words:

ASSIST ‖<A:>‖ ‖Opt: 1/6 ‖ ‖

This is the status bar. It will keep you informed of the status of the system. Right now it is saying that **ASSIST** is the current dBASE command, **A:** (or **C:** if you are using a hard disk) is the current drive, and the option currently selected is the first of six.

READ: SELECTION MESSAGE

The bottom line on the screen displays the following message:

Select a database file.

This message explains the current selection in the pull-down menu, namely **Database file**. This line will change as the selection changes, giving you information about that particular option.

READ: NAVIGATION LINE

The line between the status bar and the selection message contains the following words:

Move selection bar-||. Select-<——>. Leave menu-<—┘. Help-F1. Exit-Esc.

This is the navigation line. You navigate with the <Up>, <Down>, <Right>, and <Left> arrow keys on the right side of the keyboard. You select the option with the <Return> key.

You will use the <Left> and <Right> arrow keys to position the selection bar on an option on the menu bar. You will use the <Up> and <Down> arrow keys to position the selection bar on an option in the pull-down menu.

DO: MOVING THE SELECTION BAR

(1) Press **<Down>**. The selection bar goes down to **Format for Screen**. Notice the selection message changes accordingly.

(2) Press **<Right>**. The **Create** pull-down menu replaces the **Set Up** menu. The first option is highlighted.

(3) Press **<Down>** four times. The **Report** option in the pull-down menus is highlighted.

(4) Press **<Return>**. <Return> selects the currently highlighted option. Another box with the available disk drives appears to the right.

(5) Press **<Esc>**. <Esc> brings you back to the **Create** menu, erasing the command you have been building at the bottom of the screen on the command line.

(6) Press **<F1>**. A large box of text appears. This is a **HELP** screen. The <F1> key brings information relevant to the current situation to your attention. Read the text.

(7) Press **<Esc>**. The **HELP** box disappears and the pull-down menu reappears.

DO: BUILDING A COMMAND

(1) Press **<Right>** six times. The selection bar goes to the **Tools** option on the menu bar. The **Tools** pull-down menu is displayed on the screen and the **Set Drive** option is highlighted.

(2) Press **<Return>**. Look at the line near the bottom of the screen just above the status bar where the following has appeared on the command line:

Command: SET DEFAULT TO

With the Assistant this line displays the command you are currently building. Keep a close watch on this line as it will teach you the way to use the dBASE III PLUS command language.

(3) Press **<Down>**. The B: on the associated pull-down menu is highlighted (for hard disk hightlight A:).

(4) Press **<Return>**. You have now successfully built the command, **SET DEFAULT TO B:** (or A: for hard disk). Upon completion of the command the selection bar returns to the last option you selected.

The SET DEFAULT TO B: command makes Drive B the default drive for storing and retrieving data using dBASE III PLUS.

Immediately after powering on your computer and starting dBASE III PLUS using the Power-On Procedure, you should select or set the default drive to B: (or A: for hard disk and your practice diskette in Drive A).

READ: TWO WAYS TO GIVE A COMMAND

You have now built a dBASE III PLUS command with the aid of the Assistant by selecting options from its menus and responding to screen prompts. You can type the same command as displayed in the command line when using the Assistant (SET DEFAULT TO B:) at the dot prompt.

In your early stages of using dBASE III PLUS the Assistant helps you structure your commands. Later you may choose to use command language directly at the dot prompt because some of the more advanced commands are not available through the Assistant.

DO: THE DOT PROMPT

Press <Esc>. This exits the Assistant. The dot which appears in the lower left-hand corner of the screen is called the dot prompt. The dot prompt indicates that dBASE III PLUS is ready to accept a command. You can type dBASE commands directly at the dot prompt (called Interactive Mode) or you can use the Assistant to build dBASE commands.

Notice that the status bar still remains on the screen.

DO: THE HELP PROGRAM

At the dot prompt, you can enter commands or at any time you can call for help by entering HELP or by pressing function key F1. In either case dBASE III PLUS will display the main help menu:

Type: **.HELP <Return>**

From the main help menu, you may select one of the six options, exit HELP, or enter the name of a command for which you wish to seek information, as indicated by the key codes at the bottom of the screen:

Press <Esc> to exit the HELP menu.

dBASE III PLUS offers a tree-like arrangement of help screens and menus. You can call for help by using the HELP command in conjunction with the command for which you need help:

Type: **.HELP CREATE <Return>**

The CREATE command screen is displayed showing the proper command syntax, the command description, and some cross-reference commands.

Press <Esc> to exit the HELP screen.

DO: RE-ACTIVATING THE ASSISTANT PROGRAM

Experienced dBASE III PLUS users prefer to enter commands directly in the interactive mode, those just beginning to work with dBASE III PLUS may prefer to use the Assistant program to enter commands. You can re-activate the Assistant by performing the following command from the dot prompt:

Type: **.ASSIST <Return>**

The Assistant program is again displayed on the screen.

READ: INTERACTIVE COMMAND ENTRY

Generally, while using these learning modules, you will learn the dBASE III PL ; command language without using the Assistant program to build the commands. However, reference will be made in most cases as to the procedure for building a new command using the Assistant.

The user may invoke the Assistant or Help programs at any time as they are needed.

DO: SELECTING THE DEFAULT DRIVE

To perform dBASE III PLUS operations, the step-by-step sequence will always be presented using the Assistant program. However, if you have used dBASE II or dBASE III before and prefer to use interactive mode, the command or commands to perform a specific operation will always be presented immediately following the Assistant sequence. If you are using the Assistant, just **READ** this command.

To get an idea of how dBASE III PLUS operations are presented, repeat the steps for selecting or setting the default drive:

REMEMBER: Whenever starting dBASE III PLUS, your first command should be to set the default drive to Drive B if you are using a computer with two floppy disk drives or Drive A if you are using a hard disk:

(1) Position the selection bar on **Tools**.

(2) Position the selection bar on **Set drive** and press <Return>.

(3) Position the selection on **B:** for floppy drives or **A:** for hard drives and press <Return>.

INTERACTIVE MODE (from the dot prompt):

Type: **.SET DEFAULT TO B: <Return>**

DO: dBASE III PLUS POWER-OFF PROCEDURE

(1) Move the selection bar to **Set Up**.

(2) Press **<Down>** until the selection bar is positioned on **Quit dBASE III PLUS**.

(3) Press **<Return>**.

The DOS prompt will be displayed on the screen and you may power off the computer.

INTERACTIVE MODE (from the dot prompt):

Type: **.QUIT <Return>**

The DOS prompt will be displayed on the screen and you may power off the computer.

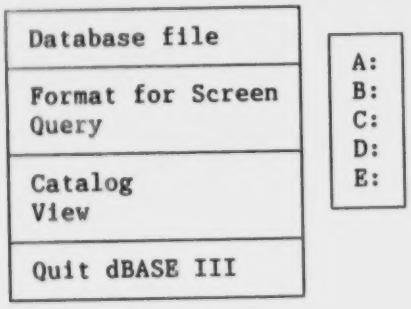
EXERCISES

Name:_____ Date:_____

Identify the following parts of the dBASE III PLUS Assistant screen:

a. Set Up Create Update Position Retrieve Organize Modify Tools

b.



c. Command: USE
d. ASSIST ‖<A:>‖ ‖Opt: 2/5 ‖ ‖
e. Move selection bar-||.Select-<——>.Leave menu-<┘.Help-F1.Exit-Esc.
f. Select a disk drive to search.

a._____

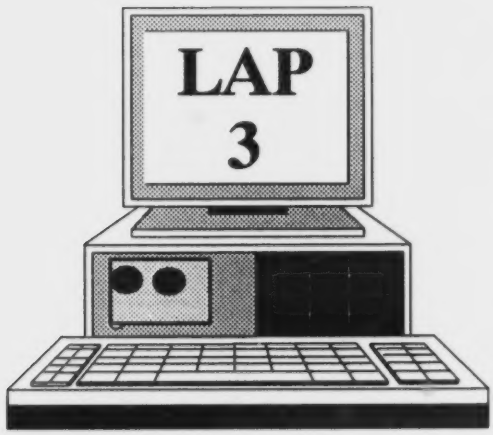
b._____

c._____

d._____

e._____

f._____


LAP
3

LESSON 3

CREATING A DATABASE

OBJECTIVES:

* Describe the CREATE command
* Create a database structure
* Select a database file for use
* Displaying and printing the database structure
* Modifying a database structure

REQUIRED MATERIALS:

* Paper and pen or pencil
* dBASE III PLUS System Diskettes 1 and 2
* A practice diskette
* IBM PC/XT, PS/II, or compatible with 2 disk drives or 1 disk drive and hard disk, and printer

READ: CREATING A DATABASE

A database must be established before you can search for answers to questions or print reports. Establishing the database tells the computer how to structure the fields and what data to store in them.

dBASE III PLUS provides English-like commands for communicating with the computer. Each command sounds like the operation that it performs. For example, the command for establishing a database is CREATE.

READ: A CASE STUDY: AL'S LIMOUSINE SERVICE

Al Cappini owns a limousine leasing service and has been leasing his limousines on a long-term, monthly basis to customers throughout the country. Al has recently purchased a microcomputer and dBASE III PLUS software to develop a computerized system for keeping track of his limos and the customers to whom they are leased.

The system will employ two files, one for the limousine information and the other for the customer information.

After several hours of thought you have designed a database structure for storing the limousine rental information. Each automobile will be represented by one record in a database containing the following information:

DATABASE NAME: LIMO

Field Name	Type	Width	Contents
LICENSE	Character	7	License Number
MAKE	Character	20	Automobile Make
YEAR	Numeric	4	Year Manufactured
RENTAL	Numeric	7	Monthly Rental Fee
CUST	Numeric	4	Customer ID

Each automobile record has five fields. Each of the fields has a field name. Each field is described in terms of its type and width. Although dBASE III PLUS provides for several data types (as previously explained), only character and numeric data types will be used for this exercise.

DO: CREATING THE DATABASE STRUCTURE

If dBASE III PLUS is not active, you should start the program and set the default drive. To create a database file structure for LIMO:

1. Position the selection bar on **Create**.

2. Position the selection bar on **Database file** and press **<Return>**.

3. Position the selection bar on **B:** if you are using floppy drives or **A:** if you are using a hard drive and press **<Return>**.

4. Enter the name of the file as **LIMO**.

INTERACTIVE MODE (from the dot prompt):

Type: **.CREATE LIMO <Return>**

When the create screen is displayed, fill in the form with the field information about the database file, LIMO. Use the screen prompts to help you advance from field to field and to correct mistakes.

The cursor is positioned under the **Field Name** column:

Type: **LICENSE** **<Return>**

The cursor advances to **Type**. Character is shown and that is what you want to press **<Return>**. The cursor advances to **Width**.

Type: **7** **<Return>**

5. Now enter definitions for the MAKE, YEAR, CUST, and RENTAL fields. To make a numeric field type, just enter N in the Type column. (You would enter D for Date and L for Logical.) Don't forget to include 2 decimal places on the RENTAL field.

   When you have finished, your cursor should be at field number 6. If any of the field information needs to be changed, use the information in the help menu at the top of the screen to move to the error and make the insertions or deletions. When you are done, save your work by pressing <Return> without typing a field name or press [Ctrl] and [End] together.

   dBASE III PLUS now creates the new file and then asks if you want to input data records now:

   Type: **N** (because we do not want to add information yet)

DO: SELECTING A DATABASE FILE

Once you've created the file structure, dBASE III PLUS stores it on the disk. Whenever you wish to perform any operation to the data in that file, such as adding, editing, deleting, or displaying, you must first USE, or open that file:

1. Position the selection bar on **Set Up**.

2. Position the selection bar on **Database file** and press **<Return>**.

3. Position the selection bar on **B:** if you are using floppy drives or **A:** (your practice diskette is in Drive A) if you are using a hard drive and press **<Return>**.

4. Position the selection bar on **LIMO.DBF** and press **<Return>**.

5. At the question, **Is the file indexed?**, press **<Return>**.

6. Notice the Status Bar now indicates that B:LIMO (or A:LIMO for hard disk systems) is the active database file.

INTERACTIVE MODE (from the dot prompt):

Type: **.USE LIMO** **<Return>**

This designates the LIMO.DBF file as the database on which to perform operations for any subsequent dBASE commands. With the LIMO database file in USE, we may now perform other operations to the database structure.

DO: DISPLAY THE DATABASE STRUCTURE

You may view the database structure on the screen at any time:

1. Position the selection bar on **Tools**.
2. Position the selection bar on **List structure** and press <Return>.
3. At the question, **Direct the output to the printer?**, enter **N**.
4. Repeat this operation and at the question, **Direct the output to the printer?**, enter **Y** to obtain a printer list.

INTERACTIVE MODE (from the dot prompt):

Type: **.DISPLAY STRUCTURE** **<Return>**

To obtain a list on the printer:

Type: **.DISPLAY STRUCTURE TO PRINT** **<Return>**

This operation has displayed the structure of the LIMO database on the screen and also printed the structure on the printer.

DO: MODIFY A DATABASE STRUCTURE

You may add, edit, or delete fields from a database structure that has already been created:

1. Position the selection bar on **Modify**.

2. Position the selection bar on **Database file** and press **<Return>**.

INTERACTIVE MODE (from the dot prompt):
Type: **.MODIFY STRUCTURE <Return>**

Make appropriate changes to the database file. If there are no changes to make, press <Return> through all the field information until the cursor advances to a blank field name. Save your work by pressing <Return> without typing a field name or [Ctrl] [End]. When the structure modifications are complete, any data that was already stored in the database will be automatically added back into the modified database. If both field length and field name are changed, data will not be added for that field and the field will remain blank.

DO: DBASE III PLUS POWER-OFF PROCEDURE

Position the selection bar to **Set Up**, position the selection bar on **Quit dBASE III PLUS**, and then press <Return>.

INTERACTIVE MODE (from the dot prompt):
Type: **.QUIT <Return>**

EXERCISES

(1) PERSONNEL DATABASE

Complete a File Structure Planning Chart (don't include Index and Report file information), create the database structure and then print the structure on the printer for the following application:

Design a database named SALARY. This database file will be used for determining if salary discrimination exists among workers in a company. This database will consist of the following information:

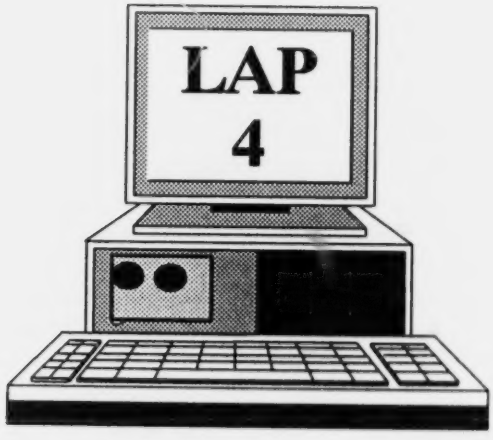
Employee First and Last Name (Width=20)
Job Title (Width=10)
Department (ACCOUNT, ADMIN, BUDGET, or MARKET)
Job Level (1, 2, 3, or 4)
Sex (Male or Female)
Years Employed
Annual Salary (xxxxx.xx)

(2) A VIDEO STORE INVENTORY DATABASE

Complete a File Structure Planning Chart (don't include Index and Report file information), create the database structure and then print the structure on the printer for the following application:

Design a database named VIDEO. This database will be used for keeping track of a Video Rental Store inventory. This database will consist of the following information:

Tape No. (a number from 1000 to 9999)
Movie Title (Width=20)
Format (VHS, BETA, or DISK)
Rating (G, PG, PG13, R, or X)
Category (Width=8)
Shelf Location (a three digit numerical location code)
List Price (xx.xx)
Unit Cost (xx.xx)


LAP
4

LESSON 4

ADDING DATA TO A DATABASE

OBJECTIVES:

* APPENDing a database file
* DISPLAYing the contents of a database file
* Selecting records from a database file
* Using conditional operators to DISPLAY a database file
* Making a back-up copy of a database file

REQUIRED MATERIALS:

* Paper and pen or pencil
* dBASE III PLUS System Diskettes 1 and 2
* A practice diskette
* IBM PC/XT, PS/II, or compatible with 2 disk drives or 1 disk drive and hard disk, and printer

READ: ADDING DATA TO A DATABASE FILE

Adding data to a database file is like adding another strip of cloth to the tail of a kite to lengthen it--data is added to the end of the database to enlarge it just as the strip of cloth is added to the end of the kite tail to lengthen it. The dBASE command to add, therefore, is APPEND. APPEND adds data at the end of a database file.

The APPEND operation tells dBASE III PLUS to add records to the database file in use. It will clear the screen and present the record format showing the fields into which you enter data and display the control key menu at the top of the screen if the menus have been set on. It will also display the record number of the next available record.

DO: APPENDING A DATABASE FILE

If dBASE III PLUS is not active, you should start the program and set the default drive.

In order to APPEND data to a database file you must have an active database:

1. Position the selection bar on **Set Up**.

2. Position the selection bar on **Database file** and press **<Return>**.

3. Position the selection bar on **B:** if you are using floppy drives or **A:** if you are using a hard drive and press **<Return>**.

4. Position the selection bar on **LIMO.DBF** and press **<Return>**.

5. At the question, **Is the file indexed?**, press **<Return>**.

6. Notice the Status Bar now indicates the B:LIMO is the active database file.

INTERACTIVE MODE (from the dot prompt):

Type: **.USE LIMO <Return>**

REMINDER: Any time you activate dBASE III PLUS you want to SET DEFAULT TO B (or A) before using dBASE commands.

The following is a list of automobiles that are currently available for lease. All but two are leased to customers. The ones that are not have a zero in the CUST field:

AL'S LIMOUSINE SERVICE--AUTOMOBILE INVENTORY

License	Make	Year	Rental	Customer
ALS-02	Cadillac	1985	1200.00	2308
ALS-04	Mercedes	1984	1800.00	3118
ALS-05	Oldsmobile	1985	800.00	0
ALS-06	Rolls-Royce	1983	1800.00	2341
ALS-09	Lincoln	1984	1200.00	3162
ALS-11	Rolls-Royce	1979	1800.00	0
ALS-13	Buick	1982	800.00	1285
ALS-14	Mercedes	1977	1200.00	2202
ALS-18	Mercedes	1968	800.00	1359
ALS-25	Rolls-Royce	1978	1200.00	957
ALS-29	Cadillac	1986	1200.00	912
ALS-33	Oldsmobile	1982	800.00	1218
ALS-38	Ford Mercury	1984	800.00	1225
ALS-39	Buick	1985	800.00	828
ALS-40	Cadillac	1984	1200.00	1898
ALS-45	Rolls-Royce	1977	1800.00	2201

To begin adding data to the LIMO database file:

1. Position the selection bar on **Update**.

2. Position the selection bar on **Append** and press **<Return>**.

INTERACTIVE MODE (from the dot prompt):
Type: **.APPEND <Return>**

Since LIMO has no data at this point, notice that the record number on the command line is **None**. Use the data from the automobile inventory, and type in the data exactly as it has been listed above, including the use of upper- and lower-case letters. Also, practice using the control key menu explanations to move from field to field and record to record in order to make any corrections to the data as needed.

You will notice that when you completely fill a field, dBASE III PLUS beeps and moves to the next field automatically. When the field contents does not fill the field, press <Return> when you've completed the entry.

After the last field of a record is entered, dBASE III PLUS automatically moves to the next record.

Notice that in entering numeric data with positions defined to the right of the decimal point, dBASE III PLUS will assign the decimal point and trailing zeros if you leave them off.

The Command line at the bottom of the screen provides information about the record position in the file as you APPEND data. Since you are APPENDing, data is added at the end of the file. EOF means end of file. **EOF/4** would mean that the record number is positioned at the end of the file and there are 4 records in the file.

3. When you have finished the entries for all of the cars, press <Return> again before entering any data in the first field of the next record and dBASE III PLUS will exit the APPEND operation.

DO: DISPLAYING THE CONTENTS OF A DATABASE FILE

Now that you have a database file with data in it, let's learn some more dBASE commands. Let's first examine all the records in the LIMO database:

1. Position the selection bar on **Retrieve**.

2. Position the selection bar on **Display** and press **<Return>**.

3. Position the selection bar on **Specify scope** and press **<Return>**.

4. Position the selection bar on **ALL** and press **<Return>**.

5. Position the selection bar on **Execute the command** and press **<Return>**.

6. After the database records are displayed on the screen you can press any key to return to Assist.

INTERACTIVE MODE (from the dot prompt): Type: **.DISPLAY ALL <Return>**

The DISPLAY command displays the current record in a database file. The DISPLAY command with the scope, ALL, displays all of the records in the file one screen at a time. DISPLAY ALL OFF does the same as DISPLAY ALL except the record number is not included. You may try this from interactive mode:

INTERACTIVE MODE (from the dot prompt): Type: **.DISPLAY ALL OFF <Return>**

Notice that the record number is not shown. The OFF is optional and suppresses the record numbers.

DO: LISTING THE CONTENTS OF A DATABASE FILE

The command, LIST, with the optional, OFF, works the same as DISPLAY ALL except that dBASE III PLUS does not pause for each screen. For instance, to LIST the database file on the printer:

1. Position the selection bar on **Retrieve**.

2. Position the selection bar on **List** and press **<Return>**. LIST assumes a scope of ALL so it is not necessary to specify the ALL scope.

3. Position the selection bar on **Execute the command** and press **<Return>**.

4. At the prompt, **Direct the output to the printer?**, type **Y**.

5. After the database records are printed on the printer you can press any key to return to Assist.

INTERACTIVE MODE (from the dot prompt): Type: **.LIST TO PRINT <Return>**

A listing of the database file, LIMO, is printed on the printer.

DO: SELECTING RECORDS FROM A DATABASE FILE

Suppose you were interested in examining the last record in the file:

1. Position the selection bar on **Position**.

2. Position the selection bar on **Goto record** and press **<Return>**.

3. Position the selection bar on **BOTTOM** and press **<Return>**.

4. Position the selection bar on **Retrieve**.

5. Position the selection bar on **Display** and press **<Return>**.

6. Position the selection bar on **Execute the command** and press **<Return>**.

7. After the database record is displayed on the screen you can press any key to return to Assist.

INTERACTIVE MODE (from the dot prompt): Type: **.GO BOTTOM <Return>** Type: **.DISPLAY <Return>**

You could think of your database file with a pointer that moves up and down the file at your command, selecting various records.

After this command the pointer would be positioned as follows:

Record No.	License	Make	Year	Rental	Customer
0001	ALS-02	Cadillac	1985	1200.00	2308
0002	ALS-04	Mercedes	1984	1800.00	3118
0003	ALS-05	Oldsmobile	1985	800.00	0
0004	ALS-06	Rolls-Royce	1983	1800.00	2341
. . . . .					
0015	ALS-40	Cadillac	1984	1200.00	1898
——>0016	ALS-45	Rolls-Royce	1977	1800.00	2201

GO or GOTO will move the pointer where you want it to go--in this case, to the last record in the file. DISPLAY (without the scope specification, ALL) displays the current or active record--that is, the record at where the pointer is placed by you.

DO: SETTING THE RECORD POINTER TO THE TOP OF FILE

To move the pointer to the first record in the database file, or the top of the file:

1. Position the selection bar on **Position**.
2. Position the selection bar on **Goto record** and press **<Return>**.
3. Position the selection bar on **TOP** and press **<Return>**.
4. Position the selection bar on **Retrieve**.
5. Position the selection bar on **Display** and press **<Return>**.
6. Position the selection bar on **Execute the command** and press **<Return>**.
7. After the database record is displayed on the screen you can press any key to return to Assist.

INTERACTIVE MODE (from the dot prompt):

Type: **.GO TOP <Return>**

Type: **.DISPLAY <Return>**

After this command the pointer would be positioned as follows:

Record No.	License	Make	Year	Rental	Customer
——>0001	ALS-02	Cadillac	1985	1200.00	2308
0002	ALS-04	Mercedes	1984	1800.00	3118
0003	ALS-05	Oldsmobile	1985	800.00	0
0004	ALS-06	Rolls-Royce	1983	1800.00	2341
. . . . .					
0015	ALS-40	Cadillac	1984	1200.00	1898
0016	ALS-45	Rolls-Royce	1977	1800.00	2201

DO: SETTING THE RECORD POINTER AT A SPECIFIC RECORD

To move the pointer to the third record in the database file:

1. Position the selection bar on **Position**.
2. Position the selection bar on **Goto record** and press **<Return>**.
3. Position the selection bar on **RECORD** and press **<Return>**.
4. At the prompt, **Enter a numeric value**, type 3 for record number 3 and press **<Return>**.
5. Position the selection bar on **Retrieve**.
6. Position the selection bar on **Display** and press **<Return>**.
7. Position the selection bar on **Execute the command** and press **<Return>**.
8. After the database record is displayed on the screen you can press any key to return to Assist.

INTERACTIVE MODE (from the dot prompt):

Type: **.GOTO 3 <Return>**

Type: **.DISPLAY <Return>**

This makes Record 3 the current or active record:

Record No.	License	Make	Year	Rental	Customer
0001	ALS-02	Cadillac	1985	1200.00	2308
0002	ALS-04	Mercedes	1984	1800.00	3118
——>0003	ALS-05	Oldsmobile	1985	800.00	0
0004	ALS-06	Rolls-Royce	1983	1800.00	2341
. . . . .					
0015	ALS-40	Cadillac	1984	1200.00	1898
0016	ALS-45	Rolls-Royce	1977	1800.00	2201

Some other acceptable forms of the GOTO 3 command would be GO 3 or GOTO RECORD 3.

To move to the next record:

9. Position the selection bar on **Position**.

10. Position the selection bar on **Skip** and press **<Return>**.

11. At the prompt, **Enter a numeric value**, type **1** to move the record pointer forward one record.

12. Position the selection bar on **Retrieve**.

13. Position the selection bar on **Display** and press **<Return>**.

14. Position the selection bar on **Execute the command** and press **<Return>**.

15. After the database record is displayed on the screen you can press any key to return to Assist.

INTERACTIVE MODE (from the dot prompt):

Type: **.SKIP <Return>**

Type: **.DISPLAY <Return>**

This makes the next record, Record 4, the current or active record:

Record No.	License	Make	Year	Rental	Customer
0001	ALS-02	Cadillac	1985	1200.00	2308
0002	ALS-04	Mercedes	1984	1800.00	3118
0003	ALS-05	Oldsmobile	1985	800.00	0
——>0004	ALS-06	Rolls-Royce	1983	1800.00	2341
. . . . .					
0015	ALS-40	Cadillac	1984	1200.00	1898
0016	ALS-45	Rolls-Royce	1977	1800.00	2201

You may move the pointer forward 3 records in the file with SKIP 3 or you may move the pointer backwards 5 records in the file with SKIP -5. Practice moving the record pointer to various positions in the file and display the record each time.

DO: DISPLAY USING CONDITIONAL OPERATORS

The dBASE command verb, DISPLAY, can be used to display a record or records that meet a certain condition or conditions. For instance, the following command would display all "Mercedes" on the screen:

1. Position the selection bar on **Retrieve**.
2. Position the selection bar on **Display** and press **<Return>**.
3. Position the selection bar on **Build a search condition** and press **<Return>**.
4. Position the selection bar on **MAKE** and press **<Return>**.
5. Position the selection bar on **Equal to** and press **<Return>**.
6. At the prompt, **Enter a character string**, type **Mercedes**.
7. Position the selection bar on **No more conditions** and press **<Return>**.
8. Position the selection bar on **Execute the command** and press **<Return>**.
9. After the database records are displayed on the screen you can press any key to return to Assist.

INTERACTIVE MODE (from the dot prompt):

Type: **.DISPLAY FOR MAKE = "Mercedes"**

The field contents, **Mercedes**, should be enclosed in quotation marks (") in interactive mode because it consists of character data. It must also be spelled exactly as it appears in the field, including upper- and lower-case letters in order for a correct match to take place.

DO: DISPLAY FOR SELECTED FIELDS

You can also DISPLAY only certain fields of a record:

1. Position the selection bar on **Retrieve**.
2. Position the selection bar on **Display** and press **<Return>**.
3. Position the selection bar on **Construct a field list** and press **<Return>**.
4. Position the selection bar on **LICENSE** and press **<Return>**.
5. Position the selection bar on **MAKE** and press **<Return>**.
6. Position the selection bar on **Build a search condition** and press **<Return>**.
7. Position the selection bar on **YEAR** and press **<Return>**.
8. Position the selection bar on **Equal to** and press **<Return>**.
9. At the prompt, **Enter a numeric value**, type **1985**.
10. Position the selection bar on **No more conditions** and press **<Return>**.
11. Position the selection bar on **Execute the command** and press **<Return>**.
12. After the database records are displayed on the screen you can press any key to return to Assist.

INTERACTIVE MODE (from the dot prompt):

Type: **.DISPLAY LICENSE, MAKE FOR YEAR=1985 <Return>**

DO: DISPLAY FOR CONDITIONS OTHER THAN EQUALITY

You can DISPLAY using other comparisons besides equality:

1. Position the selection bar on **Retrieve**.

2. Position the selection bar on **Display** and press **<Return>**.

3. Position the selection bar on **Build a search condition** and press **<Return>**.

4. Position the selection bar on **RENTAL** and press **<Return>**.

5. Position the selection bar on **Greater than** and press **<Return>**.

6. At the prompt, **Enter a numeric value**, type 1200.

7. Position the selection bar on **No more conditions** and press **<Return>**.

8. Position the selection bar on **Execute the command** and press **<Return>**.

9. After the database records are displayed on the screen you can press any key to return to Assist.

INTERACTIVE MODE (from the dot prompt):
Type: **.DISPLAY FOR RENTAL > 1200 <Return>**

Notice that when a search is used with the DISPLAY command that a scope of ALL is assumed.

READ: CONDITIONAL OPERATIONS

dBASE III PLUS uses the following conditional operators:

=	Equal to	>	Greater than
<	Less than	<>	Not equal to
>=	Greater than or equal to	<=	Less than or equal to

DO: MAKING A BACK-UP COPY OF A DATABASE FILE

As things now stand, you have the database file, LIMO, on your disk. It might be advisable at this point to make a back-up of the LIMO database.

If dBASE III PLUS is not active, you should start the program, set the default drive, and set the editing menus on. In order to back-up a database file you must have an active database:

1. Position the selection bar on **Organize**.

2. Position the selection bar on **Copy** and press **<Return>**.

3. Position the selection bar on **B:** or (**A:** if you are using a hard disk) and press **<Return>**.

4. At the prompt, **Enter the name of the file**, type LIMOBAK and press **<Return>**.

5. Position the selection bar on **Execute the command** and press **<Return>**.

6. After the database records are copied you can press any key to return to Assist.

INTERACTIVE MODE (from the dot prompt):

Type: **.USE LIMO <Return>**

To back-up or make a copy of this database:

Type: **.COPY TO LIMOBAK <Return>**

The back-up database file, LIMOBAK.DBF, is stored on the same diskette and may be USED, just like the original.

READ: RECOVERY OF A BACKUP DATABASE FILE

Let's suppose that some disaster struck and LIMO was rendered useless. The recovery procedure consists of these commands in sequence (Do not type these commands):

```
USE LIMOBAK
COPY TO LIMO
USE LIMO
```

DO: DBASE III PLUS POWER-OFF PROCEDURE

Position the selection bar to **Set Up**, position the selection bar on **Quit dBASE III PLUS**, and then press **<Return>**.

INTERACTIVE MODE (from the dot prompt):

Type: **.QUIT <Return>**

EXERCISES

1. PERSONNEL DATABASE

You have been given a very important and secret project by the president of the company for which you work. The president has asked you to analyze personnel information to see if there is any pattern of discrimination. Here is the personnel and salary data for the company:

NAME	TITLE	DEPT.	JOB	SEX	YRS	SALARY
BASIC, BRENDA	CLERK II	ADMIN	2	F	4	24,000
BEAUTY, BETTY	CLERK III	ADMIN	3	F	5	28,000
BRAINY, BOB	ANALYST I	BUDGET	2	M	3	20,000
CRISPIN, SUE	CLERK I	MARKET	1	F	3	18,500
DAINTY, DEB	ANALYST I	BUDGET	2	F	4	21,000
DEFECTA, DAN	ASST TO CEO	ADMIN	4	M	18	50,000
DIEHL, DAVE	DIR OF PLAN	ADMIN	3	M	5	36,000
DOOZER, DARREL	ADMIN ASST	ADMIN	3	M	4	30,000
EASY, EZRA	ACCOUNTANT	BUDGET	3	M	15	36,500
ESTES, SAL	CLERK I	BUDGET	1	M	4	19,000
ESTES, SUE	SECRETARY	BUDGET	1	F	1	15,000
EVANS, EDWARD	VP MARKET	MARKET	4	M	5	45,000
GERIG, CHRIS	MKT RESEARCH	MARKET	3	M	4	28,000
GOPHER, GUS	CLERK II	MARKET	2	M	4	22,000
GREEN, TOM	TREASURER	BUDGET	4	M	8	46,000
GREY, JANE	ANALYST I	BUDGET	2	F	8	26,000
HINES, GREG	CEO & BOSS	ADMIN	4	M	12	55,000
HURRY, HARRY	MKT RESEARCH	MARKET	3	M	6	32,000
HUSTLER, HILDA	SECRETARY	ADMIN	1	F	1	17,000
ICY, SUE	MKT RESEARCH	MARKET	3	F	12	36,000
JUNIOR, JIM	ADMIN ASST	ADMIN	2	M	5	26,000
LAMA, DOLLY	ADMIN ASST	ADMIN	1	F	3	19,500
LOGO, LUCY	RES ASST	MARKET	2	F	12	28,000
MACHO, MAN	CLERK I	BUDGET	1	M	2	17,000
MOON, MISSY	CLERK I	ADMIN	1	F	2	19,000
NAY, BOB	ACCOUNTANT	BUDGET	3	M	9	33,000
PICKRELL, TOM	DIR OF ACCT	BUDGET	4	M	4	38,000
SLOW, SARGE	CLERK II	BUDGET	2	M	5	24,000
SNOOZY, SAM	ANALYST II	BUDGET	3	M	5	29,000
STARR, LINDA	ANALYST II	BUDGET	3	F	10	34,000

It seems that one of the company's long-term employees, Linda Starr, has filed a salary discrimination suit against the company with the Department of Labor. This is surprising since she is well paid and appreciated by her superiors in the company. She claims that women's salaries are almost always below those of comparable men's salaries. However, what makes this claim difficult to figure out is that there are three departments and four job classifications.

You are to use dBASE III PLUS to store the data which is to be analyzed in future projects.

(1) Complete a File Structure Planning Chart (don't include Index and Report file information) for a database named SALARY.
(2) Create the database file structure.
(3) Add the personnel data to the database file.
(4) Obtain a complete list on the printer.

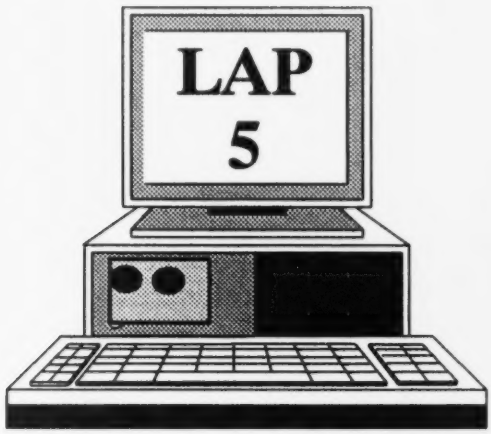
2. VIDEO STORE INVENTORY

A video tape rental store has opened for business and they want to use dBASE III PLUS to keep track of their store inventory of rental video tapes. Here is some of the video store inventory:

NO.	TITLE	FORMAT	RATING	CATEGORY	SHELF	LIST	COST
1011	Rear Window	BETA	PG	Horror	2	29.95	19.95
1214	Swing Shift	VHS	PG13	Comedy	5	45.95	32.95
1197	Urban Cowboy	VHS	R	General	8	49.95	34.95
1127	Fast Break	VHS	PG	Comedy	5	39.95	29.95
1045	Footloose	BETA	PG	Comedy	5	34.95	24.95
1123	Spies Like Us	VHS	PG13	Comedy	5	45.95	32.95
2134	Out of Africa	VHS	PG	Drama	1	39.95	29.95
1388	All of Me	VHS	PG	Comedy	5	39.95	29.95
1995	Dark Alley	BETA	R	Drama	1	29.95	19.95
1561	Side Show	BETA	R	General	8	29.95	19.95
1492	Red Dawn	VHS	R	Drama	1	34.95	24.95
1688	Survivor	VHS	PG13	Horror	2	29.95	19.95
1777	Dead Center	VHS	PG13	Drama	2	39.95	29.95
1845	Monte Walsh	VHS	PG	Western	3	29.95	19.95
2322	Gunslingers	BETA	G	Western	3	29.95	19.95
1386	On Target	VHS	PG13	Drama	2	34.95	24.95
1894	Sweet Dreams	VHS	G	General	8	29.95	19.95
1313	Sleeping Beauty	BETA	G	Children	4	29.95	19.95
1675	Howard the Duck	VHS	PG	Comedy	5	45.95	32.95
1592	Bluebeard	VHS	PG	Drama	2	39.95	29.95

You are to use dBASE III PLUS to store the data which is to be analyzed in future projects.

(1) Complete a File Structure Planning Chart (don't include Index and Report file information) for a database named VIDEO.
(2) Create the database file structure.
(3) Add the video inventory data to the database file.
(4) Obtain a complete list on the printer.


LAP
5

LESSON 5

MODIFYING DATA IN A DATABASE

OBJECTIVES:

* Modifying a database file using EDIT and BROWSE
* Marking records for deletion using DELETE
* RECALLing records marked for deletion
* Deleting records using a search condition
* Review sequence for creating and updating a database file

REQUIRED MATERIALS:

* Paper and pen or pencil
* dBASE III PLUS System Diskettes 1 and 2
* A practice diskette
* IBM PC/XT, PS/II, or compatible with 2 disk drives or 1 disk drive and hard disk, and printer

READ: MODIFYING DATA IN A DATABASE FILE

dBASE III PLUS offers several ways to change data in an existing database file. After looking over your file of data you find some corrections:

ALS-05	The fee should be $1,000.00
ALS-11	1979 should be 1969
ALS-18	The customer ID should be 1356
ALS-38	Ford Mercury should just be Mercury

In addition, there are four more limos to add to the file:

ALS-47	1977 Ford to rent for $1000.00 per month
ALS-48	1955 Bentley to rent for $1,200.00 per month
ALS-49	1968 Riviera to rent for $800.00 per month
ALS-52	1969 Toronado to rent for $800.00 per month

None of these cars is rented, so the customer ID number is zero in each case.

Let's make the corrections first, then add the new records. You will use the EDIT command to make these corrections. If you know the record number of the record to be edited, you can simply type EDIT 6 (if the record number was 6).

dBASE III PLUS will then present record 6 (or whatever record number you specify) in exactly the same manner as when you added records to the file with the APPEND command. Then, using the instructions in the control key menu, you can move about the record replacing erroneous information.

DO: SELECT THE DATABASE FILE

If dBASE III PLUS is not active, you should start the program, and set the default drive. In order to EDIT data in a database file you must have an active database:

1. Position the selection bar on **Set Up**.
2. Position the selection bar on **Database file** and press **<Return>**.
3. Position the selection bar on **B:** (or **A:** if you are using a hard disk) and press **<Return>**.
4. Position the selection bar on **LIMO.DBF** and press **<Return>**.
5. At the prompt, **Is the file indexed?**, press **<Return>**.

INTERACTIVE MODE (from the dot prompt):
Type: **.USE LIMO <Return>**

DO: RETRIEVE AND EDIT THE RECORD

You must first identify the record number of the record to be modified in order to effectively use the EDIT command:

1. Position the selection bar on **Retrieve**.
2. Position the selection bar on **List** and press **<Return>**.
3. Position the selection bar on **Build a search condition** and press **<Return>**.
4. Position the selection bar on **LICENSE** and press **<Return>**.

5. Position the selection bar on **Equal To** and press **<Return>**.

6. At the prompt, **Enter a character string**, type **ALS-11** and press **<Return>**.

7. Position the selection bar on **No more conditions** and press **<Return>**.

8. Position the selection bar on **Execute the command** and press **<Return>**.

9. At the prompt, **Direct the output to the printer?**, press **<Return>**.

INTERACTIVE MODE (from the dot prompt):
Type: **.LIST FOR LICENSE = "ALS-11" <Return>**

Notice the record number. Now you must position the pointer when using EDIT from Assist.

11. Position the selection bar on **Position**.

12. Position the selection bar on **Goto Record** and press **<Return>**.

13. Position the selection bar on **RECORD** and press **<Return>**.

14. At the prompt, **Enter a numeric value**, type the record number (for example, 6) and press **<Return>**.

15. Position the selection bar on **Update**.

16. Position the selection bar on **Edit** and press **<Return>**.

INTERACTIVE MODE (at the dot prompt):
Type: **. EDIT** and the record number and <Return> (for example, .EDIT 6 <Return>)

This identifies the record by the contents of one of its fields, in this case, the car's license number.

Go ahead and edit the data for ALS-11 as described above.

17. To terminate EDIT mode and save the changes:

    Type: **<Ctrl> <End>**

DO: MODIFYING A DATABASE USING BROWSE

Another method for modifying data is the BROWSE operation. BROWSE is a menu-assisted command that allows full-screen window editing and appending of the active database. Up to 17 records and as many fields as will fit on the screen are displayed at one time:

1. Position the selection bar on **Update**.

2. Position the selection bar on **Browse** and press **<Return>**.

INTERACTIVE MODE (from the dot prompt):
Type: **.BROWSE** **<Return>**

3. Notice the fields option on the control key menu:

   Type: **<Ctrl> <Home>** to access the options menu

Bottom	Goes to the bottom of the file (last record)
Top	Goes to the top of the file (first record)
Lock	Defines the number of fields at the left side of the screen that remain stationary during the browse
Menu	Turns the BROWSE control key menu on or off
Record #	Goes to a specified record number
Freeze	Edits on a single field

   You may use the control key menu commands or press [Esc] and use the cursor movement keys to move the cursor to the appropriate fields and make the changes for ALS-05, ALS-18, and ALS-38 as described above.

4. When you have finished making the changes:

   Type: **<Ctrl> <End>** to terminate BROWSE mode and save the changes

DO: APPEND THE NEW DATA

Now use the APPEND command from Assist or from the dot prompt to make the additions to the file as specified above.

READ: THE DELETE COMMAND

File maintenance activities include, in addition to adding records and changing the contents of existing records, deleting records from the database file.

The DELETE command of dBASE III PLUS doesn't actually remove records from the database, but merely marks them so that they will not be available for file processing operations and report production. Removing records permanently from a database requires an additional command.

The DELETE command marks all records within the specified scope of the command. DELETE ALL, for example, will mark every record in the database. DELETE by itself will mark only the active record:

DO: DELETING RECORDS FROM A DATABASE FILE

If dBASE III PLUS is not active, you should start the program, set the default drive, and USE the LIMO database file. In order to DELETE data in a database file you must have an active database:

1. Position the selection bar on **Update**.

2. Position the selection bar on **Delete** and press **<Return>**.

3. Position the selection bar on **Specify scope** and press **<Return>**.

4. Position the selection bar on **ALL** and press **<Return>**.

5. Position the selection bar on **Execute the command** and press **<Return>**.

INTERACTIVE MODE (from the dot prompt):
Type: **.DELETE ALL <Return>**

Notice the message saying how many records were deleted.

Now LIST the LIMO file using Assist or from the dot prompt. You'll see an asterisk (*) just to the left of the first field in every record. That's the deletion marker. Since the records are just "marked" for deletion, it is possible to "undelete" the records.

DO: RECALL THE DELETED RECORDS

The dBASE III PLUS command to "undelete" the deleted records is RECALL. RECALL will "recall" or "undelete" records that have been marked for deletion by the DELETE command.

1. Position the selection bar on **Update**.

2. Position the selection bar on **Recall** and press **<Return>**.

3. Position the selection bar on **Specify scope** and press **<Return>**.

4. Position the selection bar on **ALL** and press **<Return>**.

5. Position the selection bar on **Execute the command** and press **<Return>**.

INTERACTIVE MODE (from the dot prompt):
Type: **.RECALL ALL <Return>**

LIST the file again using Assist or from the dot prompt and you'll see that the marks are gone. The RECALL command simply reverses the DELETE command and operates with the same scope and condition parameters.

DO: DELETE RECORDS FOR A SEARCH CONDITION

You may delete selected records from a database by building a search condition rather than specifying a scope of ALL:

1. Position the selection bar on **Update**.

2. Position the selection bar on **Delete** and press **<Return>**.

3. Position the selection bar on **Build a search condition** and press **<Return>**.

4. Position the selection bar on **RENTAL** and press **<Return>**.

5. Position the selection bar on **Greater Than** and press **<Return>**.

6. At the prompt, **Enter a numeric value**, type 1200.

7. Position the selection bar on **No more conditions** and press **<Return>**.

8. Position the selection bar on **Execute the command** and press **<Return>**.

INTERACTIVE MODE (from the dot prompt):

Type: **.DELETE FOR RENTAL > 1200 <Return>**

LIST the LIMO file using Assist or from the dot prompt and you'll see a deletion marker by each record in which the RENTAL field contains a value greater than 1200.

DO: DELETE A SPECIFIC RECORD

The DELETE command will allow you to delete a specific record based on its position in the database file:

1. Position the selection bar on **Position**.
2. Position the selection bar on **Goto Record** and press **<Return>**.
3. Position the selection bar on **RECORD** and press **<Return>**.
4. At the prompt, **Enter a numeric value**, type 7 and press **<Return>**.
5. Position the selection bar on **Update**.
6. Position the selection bar on **Delete** and press **<Return>**.
7. Position the selection bar on **Execute the command** and press **<Return>**.
8. Position the selection bar on **Retrieve** and press **<Return>**.
9. Position the selection bar on **Display** and press **<Return>**.
10. Position the selection bar on **Execute the command** and press **<Return>**.

INTERACTIVE MODE (from the dot prompt):

Type: **.DELETE RECORD 7 <Return>**

Type: **.DISPLAY <Return>**

You should see that the last record has been marked for deletion. Remember that the DISPLAY command by itself shows only the active record. Similarly, the DELETE and RECALL commands by themselves operate only on the active record.

DO: RECALL ALL THE DELETED RECORDS

Now, remove all the deletion marks from the database file:

1. Position the selection bar on **Update**.
2. Position the selection bar on **Recall** and press **<Return>**.
3. Position the selection bar on **Specify scope** and press **<Return>**.
4. Position the selection bar on **ALL** and press **<Return>**.
5. Position the selection bar on **Execute the command** and press **<Return>**.

INTERACTIVE MODE (from the dot prompt):
Type: **.RECALL ALL <Return>**

6. LIST the file using Assist or from the dot prompt and be certain that no records are marked for deletion.

DO: REMOVE A RECORD FROM THE DATABASE FILE

It is decided not to lease the 1969 Toronado after all and you are asked to remove it from the file:

1. Position the selection bar on **Update**.
2. Position the selection bar on **Delete** and press **<Return>**.
3. Position the selection bar on **Build a search condition** and press **<Return>**.
4. Position the selection bar on **LICENSE** and press **<Return>**.
5. Position the selection bar on **Equal To** and press **<Return>**.
6. At the prompt, **Enter a character string**, type **ALS-52**.
7. Position the selection bar on **No more conditions** and press **<Return>**.
8. Position the selection bar on **Execute the command** and press **<Return>**.
9. Position the selection bar on **Pack** and press **<Return>**.

```
INTERACTIVE MODE (from the dot prompt):

     Type:   .DELETE FOR LICENSE = "ALS-52"   <Return>

     Type:   .PACK   <Return>
```

The first command marks the record for deletion. The second command, PACK, actually removes it from the file. Once the PACK command has executed, the record is gone forever.

If you PACK a file that had marked records in the middle, as opposed to the end where ALS-52 was in this example then dBASE III PLUS removes the records and renumbers the remaining records in sequence. So a PACK removes records marked for deletion and re-organizes the database file.

10. LIST the file using Assist or from the dot prompt and you'll see that ALS-52 is gone.

READ: CREATING AND UPDATING A DATABASE FILE--REVIEW

Now, you have learned how to create, add, change, delete, and back-up a database. Let's continue by preparing a second file that the company requires and use all these commands again to reinforce what you have already learned about dBASE III PLUS.

A customer database file needs to be added to your limousine inventory management system. A customer list has been prepared for you as shown below. You have been asked to create a file with the filename CUST.

The following is the field specifications for each record in the database:

DATABASE NAME: CUST

Field Name	Type	Width	Contents
CUST	N	4	Customer ID
LAST	C	14	Last Name
FIRST	C	12	First Name
ADDRESS	C	20	Address
CITY	C	16	City
STATE	C	2	State
ZIP	C	10	Zip Code
RENTAL	N	8	Rental Amount Billed
PAID	N	8	Amount Paid
DUE	N	8	Amount Due

The CUST field is numeric and has no decimal places. The last three numeric fields are dollars and cents fields and require 2 decimal places. RENTAL contains the total amount that SCRAM has charged the customer. PAID shows the total amount received in payment from the customer. Do not enter data in the final field, DUE.

DO: CREATING AND UPDATING A DATABASE FILE--REVIEW

Using Assist or from the dot prompt, perform the following operations to create and update the CUST database file:

(1) Create a database file named CUST using the field specifications shown above.

(2) Using the APPEND command, add the following names to the CUST database:

NO.	NAME/ADDRESS/CITY/STATE/ZIP CODE	BILLED	PAID
1218	Richie Rich 1000 Wealthy Way Los Angeles, CA 99999	8200.00	7200.00
3162	Arthur Nelson 4856 Fairmont Drive Colorado Springs, CO 89415	7200.00	4800.00
1225	William Wealth 9000 Lavish Lane Boston, MA 20152	4000.00	4000.00
912	Sam Saturate 1215 Gusher Blvd. Dallas, TX 74350	12000.00	6300.00
2201	David Derrick 100 Wildcat Way Midland, TX 85500	1800.00	0.00
3118	Freddy Famous 500 Star Street Beverly Hills, CA 90100	7200.00	3600.00
2341	Ronald Roller 2200 Rockford Road Tulsa, OK 73895	12600.00	7200.00
828	Edward Elite 7500 Exquisite Lane Fairmont, NY 01560	4800.00	4800.00
1356	Paul Powerful 9999 High-and-Mighty Washington, D.C. 21999	9600.00	8000.00
2308	Guiseppi Sardini 1025 Marquette Blvd. Chicago, IL 60158	3600.00	1200.00

1285	Frank Financier 900 Wall St. New York, NY 10099	5600.00	4100.00
957	Terry Tennis 40 Love Lane Palm Springs, CA 91027	4800.00	4800.00
2202	Doug Dealer 500 Cabana Ave. Miami, FL 40615	1200.00	0.00
1898	Belle Beauty 800 Turner St. Atlanta, GA 34567	3600.00	0.00

(3) After you have finished entering data into CUST, you find some corrections to make. Here are the changes:

Arthur Nelson should be Nelson Arthur
William Wealth lives in Providence, RI 22555
Frank Financier should be Frank Fiasco
Terry Tennis lives in Palm Beach, FL 41028
Belle Beauty married and is now Belle Bounty

Make these changes using the EDIT or BROWSE operation.

(4) Use the LIST or DISPLAY commands to review your database to be sure the data is correct.

(5) After all the data has been added and the appropriate changes have been made, list the data and structure to the printer.

(6) Use the COPY command to back-up your database to a file named, CUSTBAK.

DO: DBASE III PLUS POWER-OFF PROCEDURE

Position the selection bar to **Set Up**, position the selection bar on **Quit dBASE III PLUS**, and then press **<Return>**.

INTERACTIVE MODE (from the dot prompt):

Type: **.QUIT <Return>**

EXERCISES

1. PERSONNEL DATABASE

Use dBASE III PLUS to make changes and deletions to the SALARY database.

(1) Use BROWSE or EDIT to modify the following records.
(2) USE DELETE and PACK to remove the following records.
(3) Obtain a complete list on the printer.

Change:

DAN DEFECTA from ASST TO CEO to GEN MANAGER
MAN MACHO to MACK MACHO
SARGE SLOW to SID SLOW
SAM SNOOZY to SAM SLEAZY
BOB NAY to BOB NAPE
DAVE DIEHL from 36000 to 37000
MISSY MOON from CLERK I to CLERK II and Job 1 to Job 2
BETTY BEAUTY from CLERK III to CLERK I and Job 3 to Job 1
LINDA STAR from 10 years to 8 years
CHRIS GERIG from M to F

Delete:

EZRA EASY retired
SUE ICY transferred
SAL ESTES fired

Add:

HANK HANDY, ACCOUNTANT, BUDGET, 3, M, 1, 28000
SHARON ROSE, CLERK I, BUDGET, 1, F, 1, 14000

2. VIDEO STORE INVENTORY

Use dBASE III PLUS to make changes and deletions to the VIDEO database.

(1) Use BROWSE or EDIT to modify the following records.
(2) USE DELETE and PACK to remove the following records.
(3) Obtain a complete list on the printer.

Change:

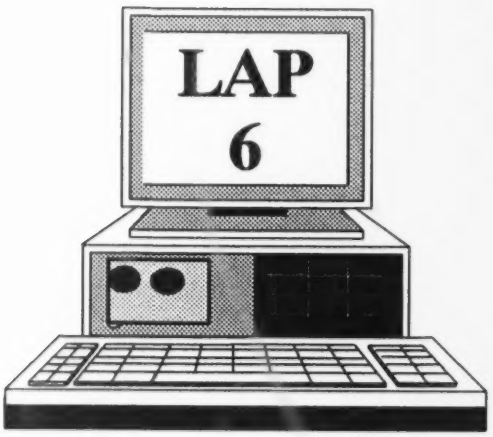
No. 1688, Survivor, from Horror to Adventure, Shelf 2 to Shelf 7
No. 1777, Dead Center, from Drama to Horror
No. 1386, On Target, from PG13 to R
No. 1011, Rear Window, from Cost 39.95 to 19.95
No. 1592, Bluebeard, from List 39.95 to 29.95, Cost 29.95 to 19.95
No. 1197, Urban Cowboy, from R to PG13
No. 1388, All of Me, from PG to PG13

Delete:

Remove all BETA format video tapes from inventory.

Add:

1598, The Hustler, VHS, R, Drama, 1, 49.95, 34.95
1649, Aliens, VHS, PG13, SciFi, 6, 39.95, 29.95
1650, Star Wars, VHS, PG, SciFi, 6, 39.95, 29.95
1741, Three Amigos, VHS, PG13, Comedy, 5, 45.95, 32.95
1888, Emerald Forest, VHS, PG, Adventure, 7, 39.95, 29.95


LAP
6

LESSON 6

REPORT AND MAILING LABEL PREPARATION

OBJECTIVES:

* Creating report specifications
* Printing a report
* Printing a report using selected parameters
* Creating label specifications
* Printing labels
* Printing labels using selected parameters

REQUIRED MATERIALS:

* Paper and pen or pencil
* dBASE III PLUS System Diskettes 1 and 2
* A practice diskette
* IBM PC/XT, PS/II, or compatible with 2 disk drives or 1 disk drive and hard disk, and printer

READ: DBASE III PLUS REPORT GENERATOR

The easiest method of generating reports is to use the internal dBASE III PLUS report generator. This is useful if the report is to be made from a single file with a minimum of logic control.

The report can be made on any portion of the file based on any criteria defined by the use, and reported columns can contain field variables, memory variables, text strings, or arithmetic expressions. The report form definition (.frm file) can be prepared earlier or at the time the report is requested. The report can be directed to the console or printer.

READ: STARTING THE REPORT GENERATOR

The command requires a form file to define the report. A data file to be reported should be selected with the USE command before you create a custom report. For example:

```
.USE Rentals INDEX Sales
```

To start a report you would enter the following:

```
.CREATE REPORT Commiss
```

This would create a report file called COMMISS.FRM.

READ: REPORT PULL-DOWN MENUS

CREATE REPORT and MODIFY REPORT are menu-assisted, full-screen commands that create and edit report form files (.FRM). The report form file is used by the REPORT FORM command to print a tabular listing of database information.

The **Report** screen is similar to the main Assist screen. A screen is displayed with a menu bar and a selection of menus on it. The **Options** pull-down menu is displayed when **Report** is first started. The bottom three lines are the familiar status bar, navigation line, and selection message line.

There are five pull-down menus displayed on the menu bar: **Options**, **Groups**, **Columns**, **Locate**, and **Exit**.

READ: THE OPTIONS MENU

On the **Options** pull-down menu a page heading of up to four lines of 60 characters each may be defined as well as the page formatting information for the report.

Default page--formatting values may be changed for page width, left margin, right margin, number of lines per page, and spacing of the lines in the report, page ejection, and page numbering.

READ: THE GROUPS MENU

The **Groups** pull-down menu directs where breaks for subtotals are to be printed as well as subtotal headings, page ejection after subtotals, and subtotals detail or summary.

Group/subtotal keys may be defined for INDEXed or SORTed files. Records with the same value on the same key field are grouped together, and subtotals are reported for numeric fields by group. A final total is calculated at the end of the report. Each subtotal group begins on a new page if the **Eject after each group/subtotal** option is chosen. Subgroups within subgroups may also be defined.

A summary report may be requested. This has only totals and subtotals for numeric fields, and field contents for the first record in each group for non-numeric fields.

READ: THE COLUMNS MENU

The **Columns** pull-down menu defines the field contents of the column, the column heading, the width of the column, and if the column is numeric then number of decimal places and whether the column should be totaled.

Contents must be defined for report fields. Each field of the report is described on a separate screen display. As many fields as can fit on the page width specified may be defined. If the number of decimal places to be displayed for a numeric field or calculation in the report is not re-defined, the default number of decimal places is used.

Even if group or subgroup totals have been defined, all numeric fields do not have to be totaled. Answer **Y** to **Total? (Y/N)** to have them totaled.

Field/column headings may be up to four lines of 60 characters each. Numeric headings are right-justified, others are left-justified.

If a field width is not specified, the default width of each report field is the width of the field definition or heading, whichever is longer. Field contents wrap around if they require more space. Summary information is displayed above the field contents.

This includes a picture of the current page layout in which X represents one character, and # or 9 represents a numeric digit.

READ: THE LOCATE MENU

The **Locate** pull-down menu will step to any previously defined column.

READ: THE EXIT MENU

The **Exit** pull-down menu will abandon the report creation operation without saving the report file or will allow you to save your report definitions in a file.

READ: REPORT FORM COMMAND

REPORT FORM prints information from the active database using the report form file created by CREATE REPORT. A report may be directed to the screen, to the printer, or an ASCII text file.

DO: SELECT THE DATABASE FILE

Designing reports with dBASE III PLUS is very easy and efficient. Report form specifications can be edited directly during the design of a report, much the same as editing database specifications when creating a database.

If dBASE III PLUS is not active, you should start the program, set the default drive, and set the editing menus on. In order to design a report form you must have an active database file:

1. Position the selection bar on **Set Up**.
2. Position the selection bar on **Database file** and press **<Return>**.
3. Position the selection bar on **B:** (or **A:** for hard disk) and press **<Return>**.
4. Position the selection bar on **CUST.DBF** and press **<Return>**.
5. At the prompt, **Is the file indexed?**, press **<Return>**.

INTERACTIVE MODE (from the dot prompt):
Type: **.USE CUST <Return>**

DO: CREATE A REPORT FILE

Using dBASE III PLUS you will enter the report specifications using a series of report specification screens, much like the dBASE III PLUS Assistant screens. A file will be created using a file name which you select and dBASE III PLUS adds the extension .FRM to the file name. To create a report named CUSTRPT1 you may use the command CREATE REPORT or MODIFY REPORT:

1. Position the selection bar on **Create**.
2. Position the selection bar on **Report** and press **<Return>**.
3. Position the selection bar on **B:** (or **A:** for hard disk) and press **<Return>**.

4. At the prompt, **Enter the name of the file:**, type **CUSTRPT1**.

INTERACTIVE MODE (from the dot prompt):
Type: **.CREATE REPORT CUSTRPT1 <Return>**

DO: REPORT FORMAT BOX

1. Type: **<F1>**

   A **Report Format** box replaces the cursor navigation menu.

2. Type: **<F1>**

   The cursor navigation menu returns.

3. Type: **<F1>**

   In the **Report Format** box you will see the format of your report as you develop it.

DO: SELECT REPORT OPTIONS

The first step requires entering the report heading and specifying details concerning the page format. The heading lines are entered in the highlighted area in the center of the screen:

1. Position the selection bar on **Options**.

2. Position the selection bar on **Page Title** and press **<Return>**.

3. At the prompt type, **AL'S LIMOUSINE SERVICE** and press **<Return>**.

4. On the next line of the report title type, **ACCOUNTS DUE REGISTER** and press **<Ctrl> <End>**.

5. Position the selection bar on **Page Width** and press **<Return>**.

6. At the prompt type, 75. Page headings are centered over the report according to the page width specification.

7. Accept all the other default values. Editing or changing report specifications is a matter of using the **<Up>** and **<Down>** arrow keys to position the selection bar. Then press **<Return>** to enter or modify specifications.

DO: SELECT GROUPS OPTIONS

Type: **<Right Arrow>**

The Groups pull-down menu appears on the screen. This pull-down menu is used to enter data concerning group and subgroup totals. This report will require no group totals so this menu can be skipped.

DO: SELECT COLUMNS OPTIONS

Type: **<Right Arrow>**

After the Group pull-down menu dBASE III PLUS presents the Columns pull-down menu to enter information for each field that is to be used as a column in the report. The contents of the field, the heading over the column, the width of the column, the number of decimal places for numeric fields, and whether the column should be totaled for numeric fields, is to be entered for each field.

The following chart shows the columns information to be entered for this report:

Field Contents	Field Header	Width	Decimal Places	Total
FIRST	FIRST NAME	12		
LAST	LAST NAME	14		
RENTAL	RENTAL AMOUNT	13	2	Y
PAID	AMOUNT PAID	11	2	Y
DUE	AMOUNT DUE	10	2	Y

1. Position the selection bar on **Contents** and press **<Return>**

2. At the prompt type, **FIRST**, and press **<Return>**. If you do not enter a valid field name, an error occurs.

OR

At the prompt press **<F10>** for a field menu. Position the selection bar on **FIRST** and press **<Return>** and then press **<Return>** again.

Notice the number 12 was automatically entered in the Width option. This is the actual field width of the field in the database file. The Width will be entered automatically based on the width of the field in the database file or the width of the column heading, whichever is greater.

3. Position the selection bar on **Heading** and press **<Return>**.

4. At the prompt type, **FIRST**, and press **<Return>** and then type **NAME** and press **<Ctrl> <End>** to save the heading entry.

5. To move to the next column press **<PgDn>** and then repeat these steps for the **LAST** field using the field information from the preceding table.

6. To move to the next column press **<PgDn>** and then repeat these steps for the **RENTAL** field using the field information from the preceding table.

   Don't forget to enter a **2** at the **Decimal places** option and a **Y** at the **Total this column** option. Notice that the width for **RENTAL** is larger than the database field width because the column heading is wider than the field width.

   Notice in the **Report Format** box at the bottom of the screen, dBASE III PLUS shows the format of the report, excluding the report headings, as of the last field specified. Column headings are shown as they will appear on the report. Underneath the headings, X's represent the width of character fields such as FIRST and LAST and 9's represent the width of numeric fields. Numeric fields such as DUE which are to be totaled are represented with #'s.

7. Now complete the report by repeating these steps for the **PAID** and **DUE** fields.

DO: SAVE THE REPORT SPECIFICATIONS

1. Position the selection bar on **Exit** and press **<Return>**.

2. Position the selection bar on **Save** and press **<Return>**.

DO: DISPLAYING THE REPORT

To display the report on the screen:

1. Position the selection bar on **Retrieve**.

2. Position the selection bar on **Report** and press **<Return>**.

3. Position the selection bar on **B:** (or **A:** for hard disk) and press **<Return>**.

4. Position the selection bar on **CUSTRPT1.FRM** and press **<Return>**.

5. Position the selection bar on **Execute the command** and press **<Return>**.

6. At the prompt, **Direct the output to the printer? Y/N**, type **N**.

INTERACTIVE MODE (from the dot prompt):

Type: **.REPORT FORM CUSTRPT1 <Return>**

DO: PRINT THE REPORT ON THE PRINTER

To print the report on the printer:

1. Position the selection bar on **Retrieve**.

2. Position the selection bar on **Report** and press **<Return>**.

3. Position the selection bar on **B:** (or **A:** for hard disk) and press **<Return>**.

4. Position the selection bar on **CUSTRPT1.FRM** and press **<Return>**.

5. Position the selection bar on **Execute the command** and press **<Return>**.

6. At the prompt, **Direct the output to the printer? Y/N**, type **Y**.

INTERACTIVE MODE (from the dot prompt):

Type: **.REPORT FORM CUSTRPT1 TO PRINT <Return>**

To print the report without the date and page number you could type:

.REPORT FORM CUSTRPT1 PLAIN TO PRINT

DO: PRINT THE REPORT FOR SELECTED CUSTOMERS

To print the report for all customers who have not made payments on their account (Paid Amount equals 0):

1. Position the selection bar on **Retrieve**.
2. Position the selection bar on **Report** and press **<Return>**.
3. Position the selection bar on **B:** (or **A:** for hard disk) and press **<Return>**.
4. Position the selection bar on **CUSTRPT1.FRM** and press **<Return>**.
5. Position the selection bar on **Build a search condition** and press **<Return>**.
6. Position the selection bar on **PAID** and press **<Return>**.
7. Position the selection bar on **Equal to** and press **<Return>**.
8. At the prompt, **Enter a numeric value**, type **0**.
9. Position the selection bar on **No more conditions** and press **<Return>**.
10. Position the selection bar on **Execute the command** and press **<Return>**.
11. At the prompt, **Direct the output to the printer? Y/N**, type **Y**.

INTERACTIVE MODE (from the dot prompt):

Type: **.REPORT FORM CUSTRPT1 FOR PAID=0 TO PRINT <Return>**

If the report needs modification or correction, the command

.MODIFY REPORT CUSTRPT1

will bring the report creation screens back into view showing the current specifications.

READ: MAILING LABELS MENUS

Creating mailing labels is much like creating reports. The Labels screen consists of two pull-down menus. The Options pull-down menu lets you provide information about the size and the format of the labels. The Contents pull-down menu lets you enter the fields from the database that are to be printed on the labels.

DO: SELECT THE DATABASE FILE

If dBASE III PLUS is not active, you should start the program, set the default drive, and set the editing menus on. In order to design a report form you must have an active database file:

1. Position the selection bar on **Set Up**.
2. Position the selection bar on **Database file** and press **<Return>**.
3. Position the selection bar on **B:** and press **<Return>**.
4. Position the selection bar on **CUST.DBF** and press **<Return>**.
5. At the prompt, **Is the file indexed?**, press **<Return>**.

INTERACTIVE MODE (from the dot prompt):
Type: **.USE CUST <Return>**

DO: CREATE A LABEL FILE

Using dBASE III PLUS you will enter the label specifications using two label specification screens, much like the dBASE III PLUS Assistant screens. A file will be created using a file name which you select and dBASE III PLUS adds the extension .LBL to the file name. To create a label file named CUSTLBL you may use either the CREATE LABEL or MODIFY LABEL command:

1. Position the selection bar on **Create**.
2. Position the selection bar on **Label** and press **<Return>**.
3. Position the selection bar on **B:** (or **A:** for hard disk) and press **<Return>**.
4. At the prompt, **Enter the name of the file:**, type **CUSTLBL1**.

INTERACTIVE MODE (from the dot prompt):

Type: **.CREATE LABEL CUSTLBL1 <Return>**

DO: SELECT LABEL OPTIONS

The first step requires telling dBASE III PLUS what the size of the label is and how many labels across the page. The standard size label is 3 1/2" x 5/16" and 1 across. On the Labels Option menu this size label is pre-defined and so we will accept this size. However we will make one slight modification by printing the labels two across:

1. Position the selection bar on **Options**.

2. Position the selection bar on **Labels across page** and press **<Return>**.

3. Type **2** and press **<Return>**.

DO: SELECT LABEL CONTENTS

1. Type: **<Right Arrow>**

   After the Options pull-down menu dBASE III PLUS presents the Contents pull-down menu to enter information for each field that is to be used on the label. The field contents of the label is to be entered.

   We want our label to look something like this:

1218	3162
Richie Rich	Nelson Arthur
1000 Wealthy Way	4856 Fairmont Drive
Los Angeles, CA 99999	Colorado Springs, CO 89415

2. Position the selection bar on **Contents** and press **<Return>**

3. Position the selection bar on field 1 and press **<Return>**. In field 1 type, **CUST** and press **<Return>**. If you do not enter a valid field name, an error occurs.

   OR

   Position the selection bar on field 1 and press **<Return>**. At the prompt press <F10> for a field menu. Position the selection bar on **CUST** and press **<Return>** and then press **<Return>** again.

4. Position the selection bar on field 2 and press **<Return>**. In field 2 type, **FIRST+LAST** and press **<Return>**. If you do not enter a valid field name, an error occurs.

OR

Position the selection bar on field 2 and press **<Return>**. At the prompt press **<F10>** for a field menu. Position the selection bar on **FIRST** and press **<Return>** and then type + and then press **<F10>** for a field menu, position the selection bar on **LAST** and press **<Return>**. Then press **<Return>** again.

5. Position the selection bar on field 3 and press **<Return>**. In field 3 type, **ADDRESS**, and press **<Return>**. If you do not enter a valid field name, an error occurs.

OR

Position the selection bar on field 3 and press **<Return>**. At the prompt press **<F10>** for a field menu. Position the selection bar on **ADDRESS** and press **<Return>** and then press **<Return>** again.

6. Position the selection bar on field 4 and press **<Return>**. In field 4 type, **CITY+" "+STATE+" "+ZIP**, and press **<Return>**. If you do not enter a valid field name, an error occurs.

OR

Position the selection bar on field 4 and press **<Return>**. At the prompt press **<F10>** for a field menu. Position the selection bar on **CITY**, press **<Return>**, then type **+" "+**. Press <F10> for a field menu, position the selection bar on **STATE**, press **<Return>**, then type **+" "+**. Press **<F10>** for a field menu, position the selection bar on **ZIP**, and press **<Return>**, then press **<Return>** again.

DO: SAVE THE REPORT SPECIFICATIONS

1. Position the selection bar on **Exit** and press **<Return>**.

2. Position the selection bar on **Save** and press **<Return>**.

DO: DISPLAYING THE LABELS

To display the labels on the screen:

1. Position the selection bar on **Retrieve**.

2. Position the selection bar on **Label** and press **<Return>**.

3. Position the selection bar on **B:** (or **A:** for hard disk) and press **<Return>**.

4. Position the selection bar on **CUSTLBL1.LBL** and press **<Return>**.

5. Position the selection bar on **Execute the command** and press **<Return>**.

6. At the prompt, **Direct the output to the printer? Y/N**, type **N**.

INTERACTIVE MODE (from the dot prompt):
Type: **.LABEL FORM CUSTLBL1 <Return>**

DO: PRINT THE LABELS ON THE PRINTER

To print the labels on the printer:

1. Position the selection bar on **Retrieve**.

2. Position the selection bar on **Label** and press **<Return>**.

3. Position the selection bar on **B:** (or **A:** for hard disk) and press **<Return>**.

4. Position the selection bar on **CUSTLBL1.LBL** and press **<Return>**.

5. Position the selection bar on **Execute the command** and press **<Return>**.

6. At the prompt, **Direct the output to the printer? Y/N**, type **Y**.

INTERACTIVE MODE (from the dot prompt):
Type: **.LABEL FORM CUSTLBL1 TO PRINT <Return>**

DO: PRINT THE LABELS FOR SELECTED CUSTOMERS

To print the labels for all customers whose monthly rental amount is greater than 1000:

1. Position the selection bar on **Retrieve**.

2. Position the selection bar on **Label** and press **<Return>**.

3. Position the selection bar on **B:** (or **A:** for hard disk) and press **<Return>**.

4. Position the selection bar on **CUSTLBL1.LBL** and press **<Return>**.

5. Position the selection bar on **Build a search condition** and press **<Return>**.

6. Position the selection bar on **RENTAL** and press **<Return>**.

7. Position the selection bar on **Greater than** and press **<Return>**.

8. At the prompt, **Enter a numeric value**, type **1000**.

9. Position the selection bar on **No more conditions** and press **<Return>**.

10. Position the selection bar on **Execute the command** and press **<Return>**.

11. At the prompt, **Direct the output to the printer? Y/N**, type **Y**.

```
INTERACTIVE MODE (from the dot prompt):

 Type: .LABEL FORM CUSTLBL1 FOR RENTAL>1000 TO PRINT  <Return>
```

If the report needs modification or correction, the command

.MODIFY LABEL CUSTLBL1

will bring the report creation screens back into view showing the current specifications.

DO: DBASE III PLUS POWER-OFF PROCEDURE

Position the selection bar to **Set Up**, position the selection bar on **Quit dBASE III PLUS**, and then press **<Return>**.

```
INTERACTIVE MODE (from the dot prompt):

     Type:   .QUIT   <Return>
```

EXERCISES

1. PERSONNEL DATABASE REPORTS

Use the Report Generator program with the database, SALARY, to prepare a report name SALRPT1 which lists the name, title, department, job, sex, years employed, and salary for all personnel in the database. Make the report heading: **PERSONNEL MASTER LIST.**

2. VIDEO STORE INVENTORY

Using the Report Generator program with the database, VIDEO, to prepare a report named VIDRPT1 which lists the number, title, format, rating category, shelf, list price, and cost for all tapes in the video inventory. Make the report heading: **VIDEO STORE INVENTORY.**

3. THE CHURCH CHOIR

Your church has recently purchased a microcomputer and dBASE III PLUS software. The choir director has asked to use the computer to maintain information about his choir members. Since you are the only choir member with computer experience, the choir director has asked you to help set up the choir member database.

After talking with the choir director, you have determined that the main problem that the choir director has is in finding qualified individuals to fill the various singing parts and roles for the musical and dramatic presentations that the choir performs. You decide that you need to know the following information about each choir member in order to provide the choir director with the information he needs:

> Member Name, Address, City, State, Zip Code, Phone No., Male or Female, Date of Birth (make this a date field), Singing Part (soprano and alto for women; tenor, bass, and baritone for men), Solo--Y/N (will or won't sing solo; make this a logical field), Musical Instrument (or NONE if doesn't play one), Date Joined the Choir (make this a date field)

(1) Complete a file structure planning chart. Create a database named, CHOIR, and add the choir member information from the following page for all members to the database.

(2) After all member information has been added to the database, obtain a print of the database structure by displaying the structure to the printer (.DISPLAY STRUCTURE TO PRINT).

(3) Using the Report Generator program, prepare a report name CHOIRLST which lists name, phone number, and singing part for all the members in the choir and print the report.

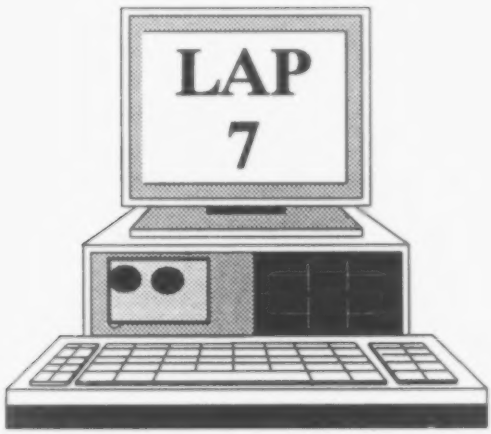
(4) Anna Treble, who was to sing a solo part in next Sunday's choir presentation, has suddenly developed laryngitis. The choir director needs a list of potential replacements as soon as possible. The solo part that Anna is singing requires a soprano voice. Modify the CHOIRLST report to include the name, phone number, singing part, and solo field for all soprano voices in the choir.

(5) Using the Label Generator program, prepare a label form named CHOIRLBL which prints name, address, city, state, and zip code on 2 across labels for all the members in the choir and print the labels.

The following are currently members of the choir:

1. Anna Treble, 2302 Park Avenue, Tulsa, OK, 74106, 918/945-6784
   Female, 03/30/42, Soprano, Piano, 01/13/75, Yes
2. Elizabeth Hansen, 633 Lee Street, Tulsa, OK, 74107, 918/946-7891
   Female, 10/13/53, Soprano, None, 12/10/82, No
3. Frank Figaro, 1232 W. Grant Ave., Tulsa, OK, 74105, 918/946-3421
   Male, 04/28/39, Bass, None, 03/25/74, Yes
4. Fred Falsetto, 1502 Warner Ave., Tulsa, OK, 74105, 918/947-4587
   Male, 11/30/58, Tenor, Saxophone, 10/12/85, Yes
5. Laura Largo, 87 Jefferson Ave., Tulsa, OK, 74106, 918/945-3410
   Female, 07/05/52, Alto, Clarinet, 05/14/80, No
6. Raul Sanchez, 6362 Wilshire Blvd., Tulsa, OK, 74104, 918/944-4433
   Male, 08/22/45, Baritone, Tuba, 12/10/82, Yes
7. David Charles, 179 N. Church, Tulsa, OK, 74106, 918/946-1029
   Male, 02/24/50, Tenor, Trumpet, 11/30/79, Yes
8. Frank Foster, 1604 Second St., Tulsa, OK, 74105, 918/947-8539
   Male, 03/18/53, Bass, None, 03/19/81, No
9. Janice Hughes, 1202 Vrain, Tulsa, OK, 74106, 918/946-0194
   Female, 10/29/54, Alto, Piano, 03/01/82, No
10. Michael Demarco, 616 Second Ave., Tulsa, OK, 74104, 918/946-6636
    Male, 03/20/58, Tenor, Tuba, 04/05/82, Yes
11. John Bennett, 1129 Vail Ave., Tulsa, OK, 74105, 918/944-2398
    Male, 03/24/57, Bass, None, 12/29/81, No
12. B.B. Massa, 1290 Railroad Drive, Tulsa, OK, 74106, 918/945-3393
    Male, 02/03/48, Baritone, Violin, 09/08/79, Yes
13. James Jaeger, 3997 Mission Blvd., Tulsa, OK, 74104, 918/946-1039
    Male, 03/10/59, Bass, Saxophone, 05/30/80, No
14. Virginia Howard, 3309 West St., Tulsa, OK, 74106, 918/944-4532
    Female, 04/19/50, Soprano, Flute, 03/30/82, Yes

15. Albert Bernier, 200 Peach Road., Tulsa, OK, 74104, 918/945-0192
    Male, 03/21/57, Tenor, None, 01/19/79, No
16. Margaret Ennis, 113 E. 41st St., Tulsa, OK, 74105, 918/944-9987
    Female, 02/15/35, Soprano, None, 10/19/72, Yes
17. Helen Freeman, 1214 State St., Tulsa, OK, 74105, 918/944-0001
    Female, 07/03/52, Alto, Clarinet, 05/30/80, No
18. Claude Malinowski, 1225 Perry Ave., Tulsa, OK, 74104, 918/942-2121
    Male, 11/10/41, Baritone, French Horn, 09/24/75, Yes
19. Elliott Goetz, 317 12th Ave., Sand Springs, OK, 74108, 918/564-3478
    Male, 12/22/58, Tenor, Guitar, 05/22/84, Yes
20. Wilma Pischke, 2302 Montgomery St., Tulsa, OK, 74105, 918/945-9273
    Female, 04/12/39, Soprano, None, 02/12/68, Yes
21. Martin White, 115 Melrose St., Sand Springs, OK, 74108,918/564-2348
    Male, 06/23/48, Bass, None, 12/10/79, No
22. Henrietta Bell, 230 Washington Dr., Tulsa, OK, 74106, 918/945-6678
    Female, 09/13/37, Alto, Organ, 04/17/69, Yes
23. Gertrude Gustafson, 450 Lindell Ave., Tulsa, OK, 74105,918/946-2235
    Female, 03/25/32, Alto, Harmonica, 01/18/70, Yes
24. Allen Hunter, 425 30th Ave., Tulsa, OK, 74106, 918/945-9980
    Male, 06/22/49, Bass, None, 10/12/81, Yes
25. Darryl Halvorsen, 163 Madison St., Tulsa, OK, 74104, 918/944-2389
    Male, 07/12/62, Tenor, None, 02/03/84, No


LAP
7

LESSON 7

MEMORY VARIABLES, OPERATIONS, AND FUNCTIONS

OBJECTIVES:

* Define and use memory variables to perform calculations
* Replacing data in a database file
* Define and use operators in dBASE commands
* Describe and use string functions
* Describe and use date functions

REQUIRED MATERIALS:

* Paper and pen or pencil
* dBASE III PLUS System Diskettes 1 and 2
* A practice diskette
* IBM PC/XT, PS/II, or compatible with 2 disk drives or 1 disk drive and hard disk, and printer

READ: MEMORY VARIABLES

Memory variables are named data items that are independently stored outside of the database structure. They provide a convenient means of temporary storage when created from the keyboard. For example, the contents of a field might be summed, and the result stored as a memory variable. The contents of the memory variable could then be directly used in further calculations. They are indispensable in programming applications where they provide the means for controlling the program.

Memory variables are similar to and may be used interchangeably with database fields in many applications (wherever the word "expression" is used in the command syntax in the Learning and Using dBASE III PLUS manuals).

Memory variable names can be up to 10 characters long. They can contain letters, numbers, and underscores. They must begin with a letter and cannot contain embedded blank spaces.

There are no restricted names for memory variables. However, problems may arise when a variable has the same name as a field in the active database file. Fields take precedence over memory variables when there is an ambiguity. An ambiguity can be resolved by preceding the memory variable name with "M".

Memory variables are normally used for temporary storage of data. However, they can be saved in a disk (.MEM) file by using the SAVE command. Variables can be recalled from a memory file by means of the RESTORE command.

Memory variables can be created by any of the following dBASE III PLUS commands:

ACCEPT, COUNT, PARAMETERS, SUM, AVERAGE, INPUT, STORE, WAIT

Variables are normally changed by overwriting existing variables with the same name. The only command that can actually edit the content of an existing variable is @..GET.

READ: MEMORY VARIABLE TYPES

Memory variables resemble fields. The rules for naming memory variables are the same as the rules for naming fields. Memory variable types are the same as field types with one exception. There are four kinds of memory variables: character, date, numeric, and logical. There is no memo variable type.

There can be 256 active memory variables whose combined size cannot exceed 6,000 bytes.

READ: CHARACTER VARIABLES

Character variables are normally used to store character strings. However a binary sequence may be stored to a character string by means of the CHR function. Character variables may contain as many as 254 characters. The total memory space used by a character variable is its length in bytes plus two bytes.

READ: DATE VARIABLES

Date variables are used to store dates. The size of a date variable is always eight bytes. The total memory requirement is nine bytes. Date variables are validated whenever they are entered or changed. Date variables are presented in the **MM/DD/YY** (American) format. They may be presented in other formats by use of the SET DATE and SET CENTURY commands.

Date variables may be created from other date variables. The CTOD function must be used to create a date variable from the keyboard. A date may be subtracted from another date. The result is a number (in days between the dates). A number (representing the number of days) may be added to or subtracted from a date. The result will be a date.

READ: NUMERIC VARIABLES

Numeric variables are used to store numbers that may be used in calculations. The accuracy of numeric variables extends to 15.9 digits. This means that the most significant 15 digits are reliable (not counting the decimal point). This is reduced to 13 digits for comparisons of non-zero numbers.

READ: LOGICAL VARIABLES

Logical variables are stored as True (.T.) or False (.F.). The size of a logical variable is always one byte. It requires a total of two bytes of memory. Logical variables will accept T, t, Y, or y for True and F, f, N, or n for False. When created from the keyboard, a logical value must be delimited by periods.

For example:

```
STORE .T. TO mlogic.
```

READ: SPECIAL CONSIDERATIONS

Memory variables are particularly useful when writing dBASE III PLUS programs.

Memory variables used in programs are subject to special considerations. When a program terminates, all memory variables created by that program are released unless they have been declared PUBLIC. The release of variables may be prevented by use of the PUBLIC command. Any variables identified as PUBLIC must be explicitly erased by means of the RELEASE command.

Two memory variables may not have the same name. The PRIVATE command is used to hide variables created by a higher-level program from a subroutine. This makes it possible for the subroutine variables to have the same names as the identified higher-level variables.

The principal commands associated with memory variables are:

ACCEPT, AVERAGE, COUNT, CLEAR MEMORY, DO, INPUT, PARAMETERS, PRIVATE, PUBLIC, RELEASE, RESTORE, SAVE, STORE, SUM, WAIT

DO: START DBASE III PLUS

If dBASE III PLUS is not active, you should start the program, set the default drive, and set the editing menus on. When the Assist program appears on the screen, press **<Esc>** because you will not need Assist for this module.

DO: USING MEMORY VARIABLES TO PERFORM CALCULATIONS

dBASE III PLUS uses the following arithmetic operators:

*	Multiplication
/	Division
+	Addition
-	Subtraction
** or ^	Exponentiation
()	Grouping

These operators can be used to calculate a value to be placed in a field based on the contents of another field or based on constants provided in the calculation expression. You'll come to that in just a moment.

In dBASE III PLUS the command, ?, can be interpreted as "What is ...?" or "Print":

Type: **.? 6*3 <Return>**

dBASE III PLUS will respond with 18. Now try this:

Type: **.? 5+10+15/3 <Return>**

Type: **.? (5+10+15)/3 <Return>**

dBASE III PLUS responds to the first calculation with 20 and to the second calculation with 10. Expressions are evaluated in dBASE III PLUS by performing multiplication and division first, then addition and subtraction. However, expressions in parentheses will always be performed first, from the inner-most to the outer-most parentheses.

In the above commands, you specified the values as constants in the expression itselt. dBASE III PLUS provides a means to store both numeric and string constants in memory by use of memory variables. To place a value in a memory variable, use the STORE command:

Type: **.STORE 5 TO A <Return>**

Type: **.STORE 10 TO B <Return>**

Type: **.STORE 15 TO C <Return>**

Type: **.STORE "The average is" TO D <Return>**

Type: **.? D,(A+B+C)/3**

dBASE III PLUS prints out a line that looks something like this (the dBASE command, ?, means "What is" or "Print"):

The average is 10

dBASE III PLUS can list all of the memory variable names you have defined and each one's contents.

Type: **.DISPLAY MEMORY <Return>**

The command RELEASE eliminates memory variables:

Type: **.RELEASE D <Return>**

Type: **.DISPLAY MEMORY <Return>**

You will see that D is no longer a memory variable.

Type: **.RELEASE ALL <Return>**

Type: **.DISPLAY MEMORY <Return>**

The specification ALL eliminates all memory variables from dBASE III PLUS memory and frees this space for other memory variables.

DO: REPLACING DATA IN A DATABASE WITH CALCULATED INFORMATION

Let's return for a minute to performing calculations and we will make use of memory variables again in just a few steps to help us perform some calculations.

You want to compute how much all the customers owe in your CUST database. If you were using a calculator, you would subtract the payments received (PAID) from the total amount charged (RENTAL) for each customer and then put the result in DUE. This can be performed quite easily in just one command phrase using dBASE III PLUS:

Type: **.USE CUST <Return>**

Type: **.REPLACE DUE WITH RENTAL - PAID <Return>**

Type: **.DISPLAY <Return>**

You will see that the REPLACE command replaces the contents of the DUE field (zero) with the remainder of the RENTAL field minus the PAID field in the first record in the database.

What if you would like to perform this operation for all records in the database, so a scope specification needs to be included with the REPLACE command:

Type: **.REPLACE ALL DUE WITH RENTAL - PAID** **<Return>**

Now list the file and you'll see that DUE has been calculated for every record.

DO: PERFORMING CALCULATION FUNCTIONS

Now you want to know the total amount due from all the customers. This can be accomplished quite easily, this time using the powerful dBASE III PLUS calculation command, SUM:

Type: **.SUM PAID** **<Return>**

The total number of records that are summed along with the result of the calculation are displayed on the screen.

The total amount due may be summed and stored in a memory variable:

Type: **.SUM PAID TO TOTAL_DUE** **<Return>**

Type: **.? TOTAL_DUE** **<Return>**

Now suppose you wanted to know the number of customers you had stored in your file. The dBASE III PLUS command, COUNT, will tally the number of records in a database file:

Type: **.COUNT TO TOTAL_CUST**

Type: **.? TOTAL_CUST**

By now you should begin to realize some of the advantages of using a database management system. Not only do you have easy access to your data for adding and updating information, but many powerful calculations can be performed on your data quite quickly and easily.

READ: RELATIONAL OPERATORS

dBASE III PLUS provides four types of operators: mathematical, relational, logical, and string. We have previously mentioned the mathematical or arithmetic operators.

Relational operators generate logical results (for example, True or False). Relational operators can be used with either character, numeric, or date expressions. Both expressions must be of the same data type:

<	Less than
>	Greater than
=	Equal
<> or #	Not equal
<=	Less than or equal
>=	Greater than or equal
$	Substring comparison (For example, if A and B are character strings, A$B returns a logical TRUE if A is either identical to B or contained within B.)

READ: LOGICAL OPERATORS

Logical operators obtain a logical result from comparing two logical expressions:

.AND.	Logical And
.OR.	Logical Or
.NOT.	Logical Not
()	Parentheses for grouping

READ: STRING OPERATORS

String operators provide for the joining (concatenation) of more than one character strings into a single character string:

+	Concatenation operator. It is used to join two or more characters strings into a single character string.
-	Concatenation operator. It is used to join two or more characters strings into a single character string. Trailing spaces, preceding the operator, are moved to the end of the string following the operator.

READ: PRECEDENCE OF OPERATORS

Each type of operator has a set of rules that governs the order in which operations are performed. These are referred to as the order of precedence for the operator.

The relational and string operators have only one order of precedence. These operations are performed in order from left to right.

The order of precedence for mathematical operators is:

1. Unary + and - signs
2. Exponentiation
3. Multiplication and division
4. Addition and subtraction

The order of precedence for the logical operators is:

1. .NOT.
2. .AND.
3. .OR.

When several of the four types of operators are used in the same expression, the order of precedence for the types is:

1. Mathematical
2. Relational
3. Logical

All operations at the same level of precedence are performed in order from left to right. Parentheses are used to override the order in which operations are performed. Operations within the innermost nested parentheses are performed first.

DO: DBASE III PLUS COMMANDS USING OPERATORS

Let's perform some commands which make use of these various types of operators.

You want to display a list of all the rental customers who owe $1000 or more on their accounts. This would be the same as saying to list all the customers in which the DUE field is greater than or equal to $1000:

Type: **.LIST FIRST,LAST,DUE FOR DUE>=1000 <Return>**

This command will list the first name, last name, and amount due fields for all customers in which the amount in the DUE field is greater than or equal to 1000.

Now you want to see all the customers who have amounts due of $1000 or more and have not paid any money. This would be the same as saying to list all the customers in which the DUE field is greater than or equal to $1000 and in which the PAID field is equal to 0:

Type: **.LIST LAST,DUE,PAID FOR DUE>=1000 .AND. PAID=0 <Return>**

This lists all last names, amount paid, and amount due fields in which the amount in the DUE field is greater than or equal to 1000 and the amount in the paid field is equal to zero.

READ: DBASE III PLUS FUNCTIONS

Functions perform specialized operations that augment and enhance the dBASE III PLUS commands. Functions always return a value and must be used as expressions or in expressions following a command verb.

Functions could be classified according to the following categories:

> Date, Character Manipulation, Mathematical, Conversion, Specialized Tests, Identification, and Input.

There are many functions to perform many types of operations. The dBASE III PLUS Learning and Using Guide can provide complete explanations of all the functions. However, we will study a few of these functions which you will find to be quite useful in selecting parameters for preparing reports.

READ: STRING FUNCTIONS

STR is a function that converts a numeric expression to a character string. For example, if you stored the number 11.14 to a variable named NUM, the function STR(NUM,5,2) would create a character string a total of 5 character spaces of which two of them would be output as decimal places.

TRIM is a function which removes trailing blanks from a character string. This is especially useful when printing or displaying multiple character strings which are used together such as First Name and Last Name or City and State.

UPPER is the upper case function. It converts lower case letters to upper case letters. For example, if you stored "abcde" to a variable named CHAR, the function UPPER(CHAR) would return a string with the value of "ABCDE".

VAL is the string-to-numeric function. It converts numbers that are defined as characters into a numeric expression. For example, if you store "123.45" to a character variable named CHAR, the function VAL(CHAR) would return the value of the variable with a numeric value of 123.45.

DO: USING STRING FUNCTIONS

Now try some operations that make use of these string functions and concatenation:

Type: **.USE CUST** <Return>

Type: **.? FIRST+LAST** <Return>

The screen should display the first name in your CUST database file with several spaces and then the last name. A better way to display or print a first and last name is:

Type: **.? TRIM(FIRST)+" "+LAST** <Return>

This allows only the first name followed by one space and then the last name to be displayed.

You can do the same with CITY and STATE and also add a comma between them:

Type: **.? CITY+", "+STATE** <Return>

Try converting the FIRST name field and LAST name field to all upper case letters as well as TRIMming the FIRST name:

Type: **.? UPPER(TRIM(FIRST)+" "+LAST)** <Return>

The important thing in using multiple functions or calculations in an expression is that you have the parentheses in the right place and that you have an equal number of open and closed parentheses.

To see how the STR function works:

Type: **.STORE 100 TO NUM1** <Return>

Type: **.STORE 200 TO NUM2** <Return>

Type: **.? NUM1 + NUM2** <Return>

The screen should display 300 as the result of this expression. Now notice the difference if these numeric variables are converted to characters:

Type: **.? STR(NUM1,3,0) + STR(NUM2,3,0)** <Return>

This time the result displayed is 100200. The STR function converts these numeric values to character strings. The "+" means concatenation for character strings. So the result returned is the string 100 joined with the string 200 or 100200.

READ: DATE FUNCTIONS

CTOD is the character-to-date function. It converts a date that has been entered or stored as a character string to a value of date type. The format of the character string is normally assumed to be MM/DD/YY. For example, to create a date memory variable named TEST_DATE you would use the following command:

```
.STORE CTOD("01/01/87") TO TEST_DATE
```

DATE() returns the system date in the form mm/dd/yy. For example, to display the system date you would use the following command:

```
.? DATE()
```

DAY returns the numeric value of the day of the month from a date expression. For example, if you had a date variable named TEST_DATE whose value is 01/15/87 then DAY(TEST_DATE) is 15.

DOW is the day of the week function. It returns a number between 1 and 7 that represents the day of the week from a date expression with Sunday being day 1 and Saturday being day 7. For example, to display the day of the week of the system date you would use the following command:

```
.? DOW(DATE())
```

DTOC is the date-to-character function. It converts a data expression to a character expression. For example, to create a character memory variable named CHAR_DATE from the system date:

```
.STORE DTOC(DATE()) TO CHAR_DATE
```

MONTH returns a number between 1 and 12 representing the month of the year from a date expression with January being 1 and December being 12. For example, if you had a date variable named TEST_DATE whose value is 07/15/87 then MONTH(TEST_DATE) is 7.

YEAR returns the numeric value of the year from a data expression. For example, if you had a date variable named TEST_DATE whose value is 01/01/87 then YEAR(TEST_DATE) is 1987.

DO: USING DATE FUNCTIONS

Now try some operations that make use of these date functions:

First, display the system date:

Type: **.? DATE()** **<Return>**

Now display the day of the week of the system date:

Type: **.? DOW(DATE())** <Return>

Now display the day of the month of the system date:

Type: **.? DAY(DATE())** <Return>

Now display the month of the year of the system date:

Type: **.? MONTH(DATE())** <Return>

Now display the year of the system date:

Type: **.? YEAR(DATE())** <Return>

Find out the number of days between two dates:

Type: **.STORE CTOD("03/15/87") TO BEG_DATE** <Return>

Type: **.STORE CTOD("10/31/87") TO END_DATE** <Return>

Type: **.? END_DATE - BEG_DATE** <Return>

The result is 230 days.

To perform mathematical calculations using dates, you must create date variables by converting character variables to date variables using the CTOD function.

DO: DBASE III PLUS POWER-OFF PROCEDURE

Since you are not using the Assistant program it is quite simple to end the processing session from the dot prompt:

Type: **.QUIT** <Return>

EXERCISES

1. PERSONNEL DATABASE REPORTS

Perform the following operations using the report generator program and the special functions and operators presented in this unit.

(1) Print the report, SALRPT1, for all personnel who have been employed for 5 or more years and whose salary is greater than $25,000. Make the report heading:

```
PERSONNEL MASTER LIST
5 OR MORE YEARS
SALARY GREATER THAN $25,000
```

(2) Print the report, SALRPT1, for all personnel who are females and whose salary is greater than $30,000. Make the report heading:

```
PERSONNEL MASTER LIST
FEMALE PERSONNEL
SALARY GREATER THAN $30,000
```

2. VIDEO STORE INVENTORY REPORTS

Perform the following operations using the report generator program and the special functions and operators presented in this unit.

(1) Print the report, VIDRPT1, for all tapes with the category equal to western and the rating equal to G or PG. Make the report heading:

```
VIDEO STORE INVENTORY
WESTERNS RATED G OR PG
```

(2) Print the report, VIDRPT1, for all tapes with the list price not equal to 29.95.

```
VIDEO STORE INVENTORY
TAPES NOT LISTED FOR $29.95
```

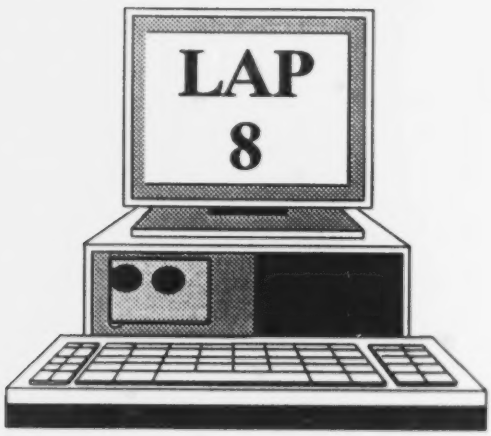
3. CHURCH CHOIR REPORTS

Perform the following operations using the report generator program and the special functions and operators presented in this unit.

(1) The church choir is forming an orchestra. Prepare a report named ORCHLST which lists the name, phone number, and instrument for all persons in the choir who play a musical instrument. (Print the report for musical instrument field is not equal to NONE.)

(2) The choir director is going to send a letter to all these persons, so prepare a mailing label form named ORCHLBL for all persons in the choir who play a musical instrument. (Print the labels for musical instrument field is not equal to NONE.)

(3) The annual Easter Drama still needs to fill a role for a male tenor approximately 30 years old. The choir director needs to hold auditions for all possible candidates for the role. Prepare a report named TENORLST with the name, phone number, sex, date of birth, and singing part for all male tenors between the ages of 25 and 35. (Print the report for the date-of-birth field is greater than or equal to a memory variable which contains a date for January 1, thirty years ago from today, and the date-of-birth field is less than or equal to a memory variable which contains a date for December 31 of the current year.)


LAP
8

LESSON 8

SORTING AND INDEXING DATABASE FILES

OBJECTIVES:

* Define sorting and indexing
* Sorting a database file
* Deleting files
* Indexing a database file
* Describe the organization of index files
* Searching a database file
* Creating an index using multiple character fields
* Creating reports using subtotals

REQUIRED MATERIALS:

* Paper and pen or pencil
* dBASE III PLUS System Diskettes 1 and 2
* A practice diskette
* IBM PC/XT, PS/II, or compatible with 2 disk drives or 1 disk drive and hard disk, and printer

READ: SORTING VS. INDEXING

Once you have your database set up, you'll want to access your data in an efficient, orderly manner. With some databases, the order in which you enter the data will be the order in which you want to get your information out. In most cases, however, you'll want it organized differently.

With dBASE III PLUS you can organize data using the SORT command or the INDEX command. The SORT command moves entire records around to set up your database in ascending or descending order on any field that you specify (name, zip code, etc.). This field is called the key.

One drawback of sorting is that you may want to access the database on one field for one application, on another field for a different application. Another drawback is that any new records added are not in order and would require a sort every time you entered data if you

wanted to maintain the order. Finding data is also relatively slow since the sorted database must be searched sequentially.

INDEXING is a way around these problems. Indexing is a method of setting up a file using only the keys that you are interested in, rather than the entire database. A key is a database field (or combination of fields) that make up the "subject" of the record. In an inventory system, the part number might be the subject, and the amount-on-hand, cost, location, etc. the descriptive fields. In a personnel database, social security numbers or employee numbers would probably make the best keys.

INDEXing creates an auxiliary file that is used in conjunction with a database file. This index (.ndx) file will cause the records to appear in the specified order.

New entries to a database are automatically added to the index file being used. Another advantage of indexed databases is the rapid location of data that you are interested in.

Specifying the desired order is accomplished by the use of a key, which can be any item of interest contained in a record. The key must be a valid character, date, or numeric expression. The use of a key will cause the records to be presented alphabetically, chronologically, or numerically.

To illustrate, the database Students is INDEXed on the Name field. The records appear in alphabetical order. Note that the record numbers are out of sequence. The record number indicates the physical position of a record in the database file. The INDEXed order is the logical order of the records.

STUDENTS.DBF

RECORD NUMBER	NAME	ROOM	GRADE
297	ADAMS	10	2
125	BENNETT	9	3
1	CALHOUN	10	2
714	DREYFUSS	15	6
. . . . .			
27	ZIGLIORE	7	5

The index file is used to find a particular record quickly when searched on a key. The structure of the index file is designed to be searched rapidly. For database files of more than a few records, the average search time is dramatically lessened with an index file, rather than without one.

There can be any number of index files associated with a database file. Up to seven index files can be open at any time. The logical order of a database file is determined by the first index file opened. The controlling index file is changed by closing all open index files and reopening them in the desired order. All open index files are updated as the content of the database changes.

An index key can be up to 100 characters long. The time it takes to create and update an index file increases with the length of the key field.

The main commands associated with indexing are:

INDEX, SET INDEX TO, FIND, SEEK, REINDEX, SET RELATION TO, USE

DO: SELECT THE DATABASE FILE

You have decided you would like the names of the customers to appear in alphabetical order by last name on the Accounts Due Register. With dBASE III PLUS sorting is very easy. The important thing to remember is that sorting produces another file. After sorting a file, you have the original, unsorted version and a new file, with its own filename, which you assign, in which the records are in sorted order.

Let's sort the customer database, CUST. If dBASE III PLUS is not active, you should start the program, set the default drive, and set the editing menus on. In order to sort a database file you must have an active database file:

1. Position the selection bar on **Set Up**.
2. Position the selection bar on **Database file** and press **<Return>**.
3. Position the selection bar on **B:** (or **A:** for hard disk) and press **<Return>**.
4. Position the selection bar on **CUST.DBF** and press **<Return>**.
5. At the prompt, **Is the file indexed?**, press **<Return>**.

INTERACTIVE MODE (from the dot prompt):

Type: **.USE CUST <Return>**

DO: SORT THE DATABASE

We will sort the database file by last name and store the sorted version of the file in a database named, CUSTLAST:

1. Position the selection bar on **Organize**.

2. Position the selection bar on **Sort** and press **<Return>**.

3. Position the selection bar on the field name **LAST** and press **<Return>**.

4. Press **<Right Arrow>** or **<Left Arrow>** to leave the Fields menu.

5. Position the selection bar on **B:** (or **A:** for hard disk) and press **<Return>**.

6. At the prompt, **Enter the name of the file:**, type **CUSTLAST**. Notice the command you have now built:

SORT ON LAST TO B:(or A:)CUSTLAST

INTERACTIVE MODE (from the dot prompt):
Type: **.SORT ON LAST TO CUSTLAST <Return>**

The field name, LAST, is the control field in the operation because you want the file to be in ascending order by the customers' last names. Ascending order for character fields is alphabetical order and ascending order for numeric or date fields is from smallest to largest.

DO: PRINT A REPORT IN SORTED ORDER

Now let's produce the Accounts Due Register in the sorted version. Remember that the REPORT FORM command operates on the active database file. CUST is still active and you wish to use CUSTLAST for the report:

1. Position the selection bar on **Set Up**.

2. Position the selection bar on **Database file** and press **<Return>**.

3. Position the selection bar on **B:** (or **A:** for hard disk) and press **<Return>**.

4. Position the selection bar on **CUSTLAST.DBF** and press **<Return>**.

5. At the prompt, **Is the file indexed?**, press **<Return>**.

6. Position the selection bar on **Retrieve**.

7. Position the selection bar on **Report** and press **<Return>**.

8. Position the selection bar on **B:** (or **A:** for hard disk) and press **<Return>**.

9. Position the selection bar on **CUSTRPT1** and press **<Return>**.

10. Position the selection bar on **Execute the command** and press **<Return>**.

11. At the prompt, **Direct the output to the printer? [Y/N]**, type **Y**.

INTERACTIVE MODE (from the dot prompt):

Type: **.USE CUSTLAST <Return>**

Type: **.REPORT FROM CUSTRPT1 TO PRINT <Return>**

DO: SORTING A DATABASE FILE IN DESCENDING ORDER

Now suppose you want the same report showing those customers who owe the most money at the top of the list. In this case you would need the report to be listed in descending order (from largest to smallest) by the amount owed.

CUSTLAST is the active database now, so let's sort it by the DUE field in descending order to CUSTDUE and run the report. You can sort a file in descending order, although not from The Assistant.

TO LEAVE ASSIST:

Press <Esc> until the dot prompt is on the screen.

Now follow the instructions for interactive mode processing.

INTERACTIVE MODE (from the dot prompt):

Type: **.SORT ON DUE TO CUSTDUE DESCENDING <Return>**

Type: **.USE CUSTDUE <Return>**

Type: **.REPORT FORM CUSTRPT1 TO PRINT** **<Return>**

The specification DESCENDING, placed at the end of the SORT command, tells dBASE III PLUS to put the new file in descending order (from largest to smallest) by the contents of the control field.

DO: DELETING DATABASE FILES

Now you have added two files to your collection--CUSTLAST and CUSTDUE. If you anticipate additional uses for these files you could keep them. However, if you added or changed the data in the original database file, CUST, these files would not be changed except by sorting the original database again. So we usually consider sorted files as temporary work files that should be removed from storage once you have finished with them.

TO START THE ASSIST PROGRAM AGAIN:

1. Type: **.ASSIST**
2. Position the selection bar on **Tools**.
3. Position the selection bar on **Erase** and press **<Return>**.
4. Position the selection bar on **B:** (or **A:** for hard disk) and press **<Return>**.
5. Position the selection bar on **CUSTLAST.DBF** and press **<Return>**.
6. When the prompt is displayed, **File has been deleted**, press the <Space Bar>.

INTERACTIVE MODE (from the dot prompt):

Type: **.DELETE FILE CUSTLAST.DBF** **<Return>**

DO: DELETING THE ACTIVE DATABASE

Since CUSTDUE is still active from the last operation we need to close or inactivate that database file before deleting.

Using Assist this is accomplished by using another database. In interactive mode this is accomplished by using no database file.

1. Position the selection bar on **Set Up**.

2. Position the selection bar on **Database file** and press **<Return>**.

3. Position the selection bar on **B:** (or **A:** for hard disk) and press **<Return>**.

4. Position the selection bar on **CUST.DBF** and press **<Return>**.

5. At the prompt, **Is the file indexed?**, press **<Return>**.

6. Position the selection bar on **Tools**.

7. Position the selection bar on **Erase** and press **<Return>**.

8. Position the selection bar on **B:** (or **A:** for hard disk) and press **<Return>**.

9. Position the selection bar on **CUSTDUE.DBF** and press **<Return>**.

10. When the prompt is displayed, **File has been deleted**, press the **<Space Bar>**.

INTERACTIVE MODE (from the dot prompt):

Type: **.USE <Return>**

Type: **.DELETE FILE CUSTDUE.DBF <Return>**

USE without a database filename closes any active database.

The DELETE FILE command is different from the DELETE command in that the DELETE FILE command will delete a complete file and can be used from any file type while the DELETE command marks a record for deletion of database files only. Notice that when you delete a file from a disk, you must use the complete file name including the extension. This is one reason it is important to understand the default extensions that dBASE III PLUS uses.

READ: INDEXING A DATABASE FILE

Instead of SORTing a database file, an index may be created based on any field or combination of fields. The field(s) used for the index is called the key field. A new file is created which contains one entry for every record in the data file. This entry holds the contents of the key field and the corresponding record number. It is similar to the index of a book which contains a list of keywords and the pages on which they may be found.

Indexed files are always in ascending order. That is, database files indexed on character fields would be in alphabetical order and database

files indexed on numeric or date fields would be in order from smallest to largest.

DO: SELECT A DATABASE FILE

Let's index the customer database, CUST. If dBASE III PLUS is not active, you should start the program, set the default drive, and set the editing menus on. In order to index a database file you must have an active database file:

1. Position the selection bar on **Set Up**.
2. Position the selection bar on **Database file** and press **<Return>**.
3. Position the selection bar on **B:** (or **A:** for hard disk) and press **<Return>**.
4. Position the selection bar on **CUST.DBF** and press **<Return>**.
5. At the prompt, **Is the file indexed?**, press **<Return>**.

INTERACTIVE MODE (from the dot prompt):
Type: **.USE CUST <Return>**

DO: CREATING AN INDEX FILE

We will index the database file by the customer id field, CUST, to an index file named CUSTID and then re-open the database file with an active index.

1. Position the selection bar on **Organize**.
2. Position the selection bar on **Index** and press **<Return>**.
3. Press **<F10>** to display the Fields menu on the screen.
4. Position the selection bar on the field name **CUST** and press **<Return>** and then press **<Return>** again in response to the completed prompt.
5. Position the selection bar on **B:** (or **A:** for hard disk) and press **<Return>**.
6. At the prompt, **Enter the name of the file:**, type **CUSTID**.

Notice the command you have now built:

INDEX ON CUST TO B:(or A:)CUSTID

7. Now position the selection bar on **Set Up**.

8. Position the selection bar on **Database file** and press **<Return>**.

9. Position the selection bar on **B:** (or **A:** for hard disk) and press **<Return>**.

10. Position the selection bar on **CUST.DBF** and press **<Return>**.

11. At the prompt, **Is the file indexed?**, type **Y**.

12. Position the selection bar on **CUSTID.NDX** and press **<Return>**.

As many as 7 indexes can be selected. The first index selected is the Master index, the one which will be used for ordering the database file.

Notice the command you have now built:

USE B:(or A:)CUST INDEX CUSTID

13. Press **<Right Arrow>** or **<Left Arrow>** to leave the Index menu and USE the database as an indexed database.

14. Position the selection bar on **Retrieve**.

15. Position the selection bar on **List** and press **<Return>**.

16. Position the selection bar on **Construct a field list** and press **<Return>**.

17. Position the selection bar on **CUST** and press **<Return>**, then **LAST** and press **<Return>**, and then **FIRST** and press **<Return>**.

18. Position the selection bar on **Execute the command** and press **<Return>**.

19. At the prompt, **Direct the output to the printer? [Y/N]**, type **N**.

INTERACTIVE MODE (from the dot prompt):

Type: **.INDEX ON CUST TO CUSTID <Return>**

Type: **.USE CUST INDEX CUSTID <Return>**

Type: **.LIST CUST,LAST,FIRST <Return>**

The LIST shows that the records are actually still in the original physical sequence, but that the file is being accessed in index order.

The index file for CUST.DBF has been saved on the disk as CUSTID.NDX, and is organized on the field named CUST. If a database file is USEd with an associated index file, all subsequent dBASE III PLUS commands will act as if the file is in index order.

READ: ORGANIZATION OF INDEX FILES

Index files are stored on the disk in a table form. A portion of the CUSTID index file could be illustrated to look something like the following table:

CUSTID.NDX

Customer No.	Record No.
828	8
912	4
957	12
1218	1
1225	3
1356	9
1898	14
2201	5
2202	13
. . . . .	

Since indexing a file does not require a complete copy to be made, substantial savings in disk storage are realized when compared to sorting. If any additions are made to the file while the index is in USE, the index will be updated automatically.

Many indexes may be created for a database file, and as many as 7 may be USEd at one time. Unfortunately, keeping many indexes active will decrease the speed of dBASE III PLUS as files are updated with the APPEND, EDIT, DELETE, and other updating commands.

DO: SEARCHING A DATABASE FILE

An additional benefit of indexing a database file is the ability to quickly search and access data in the file using the SEEK command. The contents of the indexed field are compared to an expression used with the SEEK command. If there is no match for the search request,

with the SEEK command. If there is no match for the search request, an error messages will be displayed on the screen stating that there was no find of the specified expression. If there is a match for the search request, the record pointer will be positioned at the requested record.

1. Position the selection bar on **Position**.

2. Position the selection bar on **Seek** and press **<Return>**.

3. At the prompt, **Enter an expression:**, type **1898**.

4. Position the selection bar on **Retrieve**.

5. Position the selection bar on **Display** and press **<Return>**.

6. Position the selection bar on **Execute the command** and press **<Return>**.

INTERACTIVE MODE (from the dot prompt):

Type: **.SEEK 1898 <Return>**

Type: **.DISPLAY <Return>**

The customer record for Jack Ketchum should be displayed on the screen.

When using a SEEK command, if the desired expression is a character string, it must be enclosed by quotation marks. If the desired expression is a memory variable or a numeric expression, as in the above example, it is used directly.

READ: CONCATENATION ON INDEXING

An index file can be created using a combination of fields which have been concatenated or joined together to form one larger character field.

If numeric fields are to be used in the concatenation, they must be converted to character fields by use of the STR function. If date fields are to be used in the concatenation they must be converted to character fields by use of the DTOC function.

Sometimes it is advantageous to drop the trailing blanks from character fields which are to be concatenated for an index file by using the TRIM function.

DO: CREATING AN INDEX USING MULTIPLE CHARACTER FIELDS

We shall create another index file for the CUST database-- this time indexed by last and first names. The name of the index file will be CUSTNAME.

1. Position the selection bar on **Organize**.
2. Position the selection bar on **Index** and press **<Return>**.
3. Press **<F10>** to display the Fields menu on the screen.
4. At the prompt, **Enter an index key expression:**, type **TRIM(** .
5. Position the selection bar on the field name **LAST** and press **<Return>**.
6. At the prompt, **Enter an index key expression:**, type **)+** to close the parentheses around LAST, state the concatenation function, and then press **<F10>** again.
7. Position the selection bar on the field name **FIRST** and press **<Return>** and then press **<Return>** again in response to the completed prompt.
8. Position the selection bar on B: (or A: for hard disk) and press **<Return>**.
9. At the prompt, **Enter the name of the file:**, type **CUSTID**.

   Notice the command you have now built:

   INDEX ON TRIM(LAST)+FIRST TO B:(or A:)CUSTNAME

10. Now position the selection bar on **Set Up**.
11. Position the selection bar on **Database file** and press **<Return>**.
12. Position the selection bar on **B:** (or **A:** for hard disk) and press **<Return>**.
13. Position the selection bar on **CUST.DBF** and press **<Return>**.
14. At the prompt, **Is the file indexed?**, type **Y**.
15. Position the selection bar on **CUSTNAME.NDX** and press **<Return>**.
16. Press **<Right Arrow>** or **<Left Arrow>** to leave the Index menu and USE the database as an indexed database.
17. Position the selection bar on **Retrieve**.

18. Position the selection bar on **List** and press **<Return>**.

19. Position the selection bar on **Construct a field list** and press **<Return>**.

20. Position the selection bar on **LAST** and press **<Return>**, then **FIRST** and press **<Return>**, and then **CUST** and press **<Return>**.

21. Position the selection bar on **Execute the command** and press **<Return>**.

22. At the prompt, **Direct the output to the printer? [Y/N]**, type **N**.

INTERACTIVE MODE (from the dot prompt):

Type: **.INDEX ON TRIM(LAST)+FIRST TO CUSTID <Return>**

Type: **.USE CUST INDEX CUSTNAME <Return>**

Type: **.LIST LAST,FIRST,CUST <Return>**

DO: CREATING REPORTS USING SUBTOTALS

Try this operation for extra practice using the SORT and REPORT FORM commands to create a report with subtotals (using either The Assistant program or from the dot prompt).

In order to have a report with subtotals, the database file must be either sorted or indexed by the field on which you are going to subtotal. Use the following steps to perform this operation. These steps will work for any report in which you are going to create subtotals. In this case we are going to SORT, however, you could just as well INDEX:

1. Use the database file, LIMO.

2. SORT the database file on the field, MAKE, to a database file named LIMOMAKE.

3. Use the database file, LIMOMAKE.

4. Create a report named LIMORPT1. Use all the fields in the report. Title the report: LIMOUSINE LIST SUBTOTALED BY MAKE. On the Groups screen enter the field name, MAKE, at the Group on expression prompt. At the Group heading prompt, type TOTAL BILLED.

DO: DBASE III PLUS POWER-OFF PROCEDURE

Position the selection bar to **Set Up**, position the selection bar on **Quit dBASE III PLUS**, and then press **<Return>**.

INTERACTIVE MODE (from the dot prompt):

Type: **.QUIT <Return>**

EXERCISES

1. CHURCH CHOIR REPORTS

Create the following reports using the dBASE III PLUS report generator program with sorted and indexed database files:

(1) The choir director wants to honor the choir members who have been choir members for the most number of years at the annual Church Choir Banquet. Prepare a report named YEARSLST with the name, date member joined the choir, and the number of years the member has been in the choir as of today's date for all choir members beginning with the member who has been in the choir for the most number of years to the one who has been in the choir the least number of years.

(2) The choir director also wants to recognize the oldest and youngest choir members at the banquet. Prepare a report with the name, date of birth, and the age as of today's date for all choir members beginning with the member who is the oldest to the member who is the youngest.

2. LEASE RECORDS SYSTEM

ABC Oil Company, a small independent oil company, purchases leases on tracts of land for future development of the oil reserves located on those properties. ABC Oil Company needs a computerized system for managing these leased properties due to the number of leases the company is acquiring and the complexity of maintaining all the properties in accordance with the conditions of each lease.

Oil and gas lease operators normally acquire rights to mineral properties through mineral leases rather than purchase of the property. The most common provisions of a mineral lease are as follows:

- LEASE BONUS - cash or other consideration paid to the lessor by the lessee in return for the rights to explore for minerals (oil and gas). The amount of bonus paid is a result of bargaining between the parties and is affected by factors such as proximity of the property to proven production and term of the lease.

- PRIMARY TERM - the maximum period of time allowed for the lessee to commence drilling a well. The lessor wants the oil company to drill as quickly as possible and the oil company would prefer a long period of time so that the property can be evaluated and a drilling budget set. A two or three year primary term is quite common.

- DRILLING OBLIGATION - the lease stipulates that drilling must begin within one year. If drilling does not begin by the anniversary date, the lease will terminate regardless of the primary term unless the lessee makes a specified payment to the lessor. The payment made to defer drilling activities for an additional year is called a DELAY RENTAL. The amount of delay rental is stated on a per-acre basis and is normally much smaller than the lease bonus. The delay rental is deposited to the lessor at a specified bank. A PAID-UP lease owes no delay rental.

- ROYALTY PROVISIONS - the lessor retains a royalty interest in the minerals. This interest entitles the lessor to receive free and clear of all costs a specified portion of the oil and gas produced, or value of such production. The royalty share is negotiable although historically a typical royalty on oil and gas properties is one eighth.

- PRODUCTION HOLDS LEASE - once a successful well has been drilled and commercial production is obtained, the lease remains in effect for as long as there is production without extended and indefinite interruption. If a successful well has been drilled but production has not begun due to absence of a reasonable market for the oil or gas or other extenuating circumstances, the well is SHUT-IN. Most lease contracts provide for a shut-in royalty to be paid to the royalty owner if this occurs. Generally, this amount is equal to the delay rentals amount.

(1) CREATE AND UPDATE DATABASE

Create a database file name, LSE, and add the lease information for all leases to the database. Create fields for lease no., lessor, term of lease (numeric), no. of acres, acquired, rental, and expiration date fields, delay rental per acre, bonus per acre, royalty percentage (expressed as a decimal number with at least 4 decimal places), status, county, and state. All leases should have one of the following statuses: (1) Undeveloped, (2) Paid-Up, (3) Held by Production, or (4) Shut-In (use codes UN, PU, HP, or SI).

After all lease information has been added to the database, obtain a print of the database structure by displaying the structure information to the printer. (.DISPLAY STRUCTURE TO PRINT)

(2) LEASE MASTER LISTINGS

a. Using the Report Generator program, prepare a report named LEASELST which lists the lease number, lessor, date acquired, primary term of lease, no. of acres, state, and county in order by lease number. Use the report heading: LEASE MASTER LIST

b. Print the report, LEASELST in order by location (state + county). Add the subtitle, SORTED BY LOCATION, to the report heading.

(3) LEASE EXPIRATION SCHEDULE

Using the Report Generator program, prepare a report named LEASEEXP which lists the lease number, lessor name, location, expiration date, and status in lease expiration date order. Print all lease expirations for which production has not held the lease (status is equal to UN or PU). Use the report heading: LEASE EXPIRATION SCHEDULE.

(4) LEASE RENTAL SCHEDULES

a. Using the Report Generator program, prepare a report named LEASERNT which shows rentals due for February, 1986 (MONTH function of the rental date is equal to 2 and YEAR function of the rental date is equal to 1986) for leases with rentals due (status is equal to UN). Include the lease number, county and state, lessor name, status, rental date, and rental amount (delay rental per acre x no. of acres). Total the rental amounts. Use the report heading: LEASE RENTAL SCHEDULE.

b. Print the report, LEASERNT, to list lease rentals subtotaled by month due (sort and subtotal by MONTH function of the rental date) for leases with rentals due (status is equal to UN) for the year 1986. Subtotal and total rental amount.

(5) LEASE COST SCHEDULE

Using the report Generator program, prepare a report named LEASECST which lists the total dollars spent for each lease in lease number order. Include the lease number, lessor name, delay rental amount (delay rental x no. of acres), bonus amount

(bonus per acre x no. of acres) and total cost (bonus per acre x no. of acres + delay rental per acres x no. of acres) for lease. Total all amounts for all leases. Use the report heading: LEASE COST SCHEDULE.

Use the following Lease Data:

Lease No. 100, Lessor is William Windfall, lease acquired is 2-23-85, primary term is 3 years, delay rental date is anniversary date, 80 acres in the lease, delay rental amount per acre is $20.00, bonus paid to the lessor is $100.00 per acre, shut-in royalty is equal to the delay rental amount, lessor royalty percentage is 1/8th, status of lease is undeveloped, location is Lea County, N.M.

Lease No. 200, Digger Rigger, lessor, lease acquired 6-14-85, 3 yr. term, rental is anniversary date, 40 acres, $15.00/acre rental, $50/acre bonus, shut-in equal to rental, 1/8th royalty, undeveloped, Garvin County, OK.

Lease No. 101, Dan Drilling, lessor, acquired 2-20-85, 2 yr. term, no rental, 40 acres, $200/acre bonus, 1/8th royalty, paid-up, Carter Co., Oklahoma

Lease No. 105, Fred Farmer, lessor, acquired 3-25-85, 3 yrs, rental is anniversary date, 40 acres, $25/acre rental, $150/acre bonus, shut-in royalty equal to rental, 1/8th royalty, shut-in, Mineral County, Texas.

Lease No. 102, Rick Reserves, lessor, acquired 2-25-85, 3 yrs., rental is anniversary date, 160 acres, $15/acre rental, $75/acre bonus, shut-in equal to delay rental, 1/8th royalty, undeveloped, Eddy County, N.M.

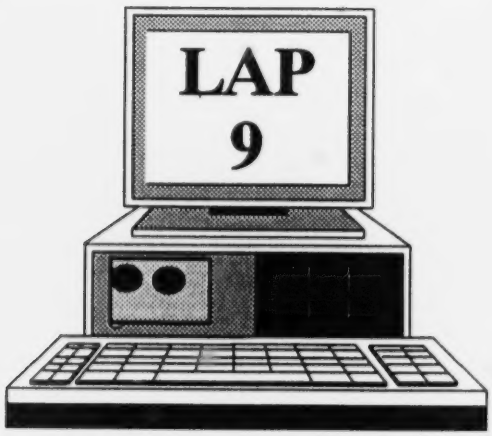
Lease No. 150, Abner Asphalt, lessor, acquired 4-10-85, 3 yrs., rental is anniversary date, 120 acres, $20/acre rental, $100/acre bonus, shut-in equal to delay rental, 1/10 royalty, undeveloped, Canadian County, Oklahoma

Lease No. 120, Roy Rancher, lessor, acquired 3-31-85, 3 yrs., rental is anniversary date, 60 acres, $10/acre rental, $80/acre bonus, shut-in equal to delay rental, 1/8 royalty, undeveloped, Armadillo County, Texas.

Lease No. 103, Paul Payout, lessor, acquired 2-27-85, 3 yrs., rental is anniversary date, 80 acres, $20/acre rental, $120/acre bonus, shut-in equal to delay rental, 3/16th royalty, held by production, Rocky County, Wyoming.

Lease No. 155, Oscar Offshore, lessor, acquired 4-12-85, 3 yrs., rental is anniversary date, 80 acres, $12/acre rental, $50/acre bonus, shut-in equal to delay rental, 1/8th royalty, undeveloped, Cajun County, Louisiana.

Lease No. 175, Harold Hillbilly, lessor acquired 5-2-85, 3 yrs., rental is anniversary date, 120 acres, $8/acre rental, $25/acre bonus, shut-in equal to delay rental, 1/16th royalty, undeveloped, Dogpatch County, Arkansas.


LAP
9

LESSON 9

DATA ENTRY FORMS

OBJECTIVES:

* Creating a screen format
* Using the screen format blackboard
* Adding graphics to a screen format
* Selecting a screen format for data entry
* Updating a database file using a screen format

REQUIRED MATERIALS:

* Paper and pen or pencil
* dBASE III PLUS System Diskettes 1 and 2
* A practice diskette
* IBM PC/XT, PS/II, or compatible with 2 disk drives or 1 disk drive and hard disk, and printer

READ: DATA ENTRY FORMS

A data entry form provides a visual way of presenting and receiving information. It can be a printed paper form, such as a job application form or an on-screen form which feeds information to a computer program. Just as you would fill in spaces in the job application form, you can fill in the blank spaces on an on-screen form. Since your form is not on paper which is filed away in a file cabinet, you can feed your collected information to your database and put it right to work.

Data entry forms provide an easy-to-understand and efficient way for adding and editing data in your database. You can create electronic on-screen forms that look like printed paper forms by designing or "painting" your form right on the screen. Once you have designed a form it can be used over and over for adding or editing data in your database.

DO: CREATE A SCREEN FORMAT

You are going to create an input form to be used in conjunction with the LIMO database. If dBASE III PLUS is not active, you should start the program, set the default drive, and set the editing menus on.

1. Position the selection bar on **Create**.

2. Position the selection bar on **Format** and press **<Return>**.

3. Position the selection bar on **B:** (or **A:** for hard disk) and press **<Return>**.

4. At the prompt, **Enter the name of the file:**, type **LIMOFORM** and press **<Return>**. dBASE III PLUS will automatically add the extension, .FMT.

5. Position the selection bar on **Set Up**.

6. Position the selection bar on **Select Database File** and press **<Return>**.

   In order to create a screen input form, you must have an active database file.

7. Position the selection bar on **LIMO.DBF** and press **<Return>**.

8. Position the selection bar on **Load Fields** and press **<Return>**.

   A form doesn't have to include all the fields in your database file, so you should mark each field you want.

9. Position the selection bar on the field name, **LICENSE**, and press **<Return>**. The symbol appears before LICENSE. For this form you want to include all the fields, so mark the remaining fields in the same way.

10. Now press or to indicate that you're finished marking fields. The field submenu will disappear and the fields you selected appear on a blank screen called the blackboard.

    The blackboard is a special screen where you can design or "paint" your form and fit the selected fields in the form.

INTERACTIVE MODE (from the dot prompt):

Type: **.CREATE SCREEN LIMOFORM <Return>**

DO: USING THE BLACKBOARD

You have now identified which fields of LIMO.DBF you want to include in LIMOFORM.FMT. Now you will layout your form on the blackboard. As you will see, some changes affect the form only, while others affect both the form and the database file structure.

Before you begin the layout process, take a look at the blackboard. The field names appear along the left edge of the screen. To the right of each field name is a highlighted area, containing symbols showing the type of character allowed in that field. This area is also called the field highlight. X's in the highlighted area indicate a character field, 9's indicate numeric and date fields, and an L indicates a logical field.

Also notice the status bar at the bottom of the screen. That status bar tells you the current location (row and column) of the cursor and provides other useful information.

1. Press **Ins** (insert key) a few times and notice what happens with the **Ins** (Insert/Overwrite) indicator on the right side of the status line.

   When **Ins** displays, you are in Insert mode. Any text you type is added to the line at the cursor location, pushing ahead what is already there.

   Initially dBASE III PLUS is in Overwrite mode. Any characters you type will write over what is already there.

   When you press **Ins** to change modes, make sure your cursor is not positioned on a field highlight. If it is, you will not change modes, but will add characters to the field highlight on which the cursor is positioned. In the same manner, pressing the **Del** (delete) key will delete characters from the field highlight.

2. In Insert mode with the cursor at the top left of the screen, press **<Return>** six times. All the fields and their corresponding text move down six lines. The blank lines provide space for adding a title to the form.

3. Insert one blank line between the LICENSE field and the MAKE field by using the <Down Arrow> key to position the cursor on MAKE and press **<Return>** one time. Now repeat this process to insert one blank line between all the fields.

4. Still in Insert mode, use the **<Spacebar>** to insert 15 spaces at the beginning of each line having a field.

5. Insert 5 more spaces between the field name and the field highlight by moving the cursor to the beginning of each field

highlight and using the **<Spacebar>** to insert 5 spaces. Repeat this process for each field highlight.

6. Add a title by using the arrow keys to move the cursor to row 2 column 22 and type, **AL'S LIMOUSINE SERVICE**. Now move to row 3 column 21 and type, **Limousine Data Entry Form**.

7. Press the **Ins** key one time to change to Overwrite mode. Now change the field label for LICENSE to LICENSE NO. by typing the word, **NO.** after the word LICENSE. Move down to the field label CUST and replace it with CUSTOMER NO. by typing **CUSTOMER NO.** over CUST. Move down to the field label RENTAL and replace it with the label, **MONTHLY RENTAL**.

   At this point your screen should look like the following:

```
                AL'S LIMOUSINE SERVICE
              Limousine Data Entry Form


        LICENSE NO.       XXXXXXX

    MAKE                XXXXXXXXXXXXXXXXXXX

        YEAR                9999

        CUSTOMER NO.        9999

    MONTHLY RENTAL      9999.99
```

DO: ADDING GRAPHICS TO THE FORM

You can draw boxes on the form to enhance your screen display.

1. Press **<F10>**. The Screen Menu Bar now appears on the screen.

2. Position the selection bar on **Options**.

3. Position the pointer on **Draw a window or line - Double bar** and press **<Return>**.

4. We are going to draw a double line box around the title. To specify the upper left corner of the box, place the cursor on row 1 column 13 and press **<Return>**.

5. Now indicate the lower right corner of the box by placing the cursor on row 4 column 55 and pressing <Return>. A double-line box should be drawn around the titles.

6. Now draw another box, this time single line. Press **<F10>** to return to the Screen Menu Bar. Position the selection bar on **Options**. Position the selection bar on **Draw a window or line-**

Single bar. Move the cursor to row 5 column 13 and press **<Return>** to specify the upper right hand corner. Move the pointer to row 15 column 55 and press return. A single-line box should be drawn around the field labels and highlights.

At this point your screen should look like the following:

```
┌──────────────────────────────────────┐
│        AL'S LIMOUSINE SERVICE        │
│        Limousine Data Entry Form     │
└──────────────────────────────────────┘

┌──────────────────────────────────────┐
│ LICENSE NO.      XXXXXXX             │
│                                      │
│ MAKE             XXXXXXXXXXXXXXXXXXX │
│                                      │
│ YEAR             9999                │
│                                      │
│ CUSTOMER NO.     9999                │
│                                      │
│ MONTHLY RENTAL   9999.99             │
└──────────────────────────────────────┘
```

7. Obtain a print of the form on the printer by using the **PrtSc** key. The graphics characters may not print on your printer in the same way it is displayed on the screen.

8. Now save the form so that you can use it any time to add or edit records by pressing **<F10>** to return to the Screen Menu Bar. Position the pointer on **Exit** and then position the pointer on **Save** and press **<Return>**.

DO: SELECT A DATA ENTRY FORM TO UPDATE A DATABASE FILE

When using a data entry form to add and edit records, you must not only select the database file, you must also select the screen format file:

1. Position the selection bar on **Set Up**.

2. Position the selection bar on **Database File** and press **<Return>**.

3. Position the selection bar on **B:** (or **A:** for hard disk) and press **<Return>**.

4. Position the selection bar on **LIMO.DBF** and press **<Return>**.

5. At the prompt, **Is the file indexed?**, press **<Return>**.

6. Position the selection bar on **Format for Screen** and press **<Return>**.

7. Position the selection bar on **B:** (or **A:** for hard disk) and press **<Return>**.

8. Position the selection bar on **LIMOFORM.FMT** and press **<Return>**.

INTERACTIVE MODE (from the dot prompt):

Type: **.USE LIMO <Return>**

Type: **.SET FORMAT TO LIMOFORM <Return>**

DO: ADDING AND EDITING USING A DATA ENTRY FORM

Now that you have selected a database file and a data entry form to update that file, let's add and edit some records:

1. Position the selection bar on **Update**.

2. Position the selection bar on **Append** and press **<Return>**.

3. Add the following record information to the data entry screen:

   ALS-60 Lincoln 1986 0 1600.00

4. Use the **Edit** function to edit the same record and change the customer number from 0 to 1505.

INTERACTIVE MODE (from the dot prompt):

Type: **.APPEND <Return>**

DO: DATA ENTRY SCREEN REVIEW

Now try to create a data entry screen using a similar design or your own design for the CUST database:

1. Design a screen named CUSTFORM for CUST.DBF using all the fields. Include a title on the form such as:

   AL'S LIMOUSINE SERVICE
   Customer Data Entry Form

2. Change the field labels to be more explanatory of the data to be input to each field.

3. Using graphic lines to enhance the looks of your data entry form.

4. Add the following data to your customer database using the new data entry form:

   1505 Robin Rocker
   9999 Heavy Metal Drive
   Los Angeles, CA 99878

READ: MORE SCREEN DESIGN FUNCTIONS

There are many other possibilities when using the screen design function. For instance, you can move fields, change the width of fields, add extra fields, delete fields, or change a field's display and entry characteristics. You should refer to the LEARNING AND USING DBASE III PLUS manual that is included with the dBASE III PLUS software package.

DO: DBASE III PLUS POWER-OFF PROCEDURE

Position the selection bar to **Set Up**, position the selection bar on **Quit dBASE III PLUS**, and then press **<Return>**.

INTERACTIVE MODE (from the dot prompt):
Type: **.QUIT <Return>**

EXERCISES

1. SCREEN DESIGNS

Perform the following applications using the screen design function of dBASE III PLUS. Include titles, new field labels, and graphics lines on each data entry screen.

After the screen design is complete and saved, select the corresponding database file and screen format and use the append operation to display the screen format on the computer screen. Use the PrtSc key to obtain a print of the screen format on the printer.

(1) PERSONNEL DATABASE SCREEN DESIGN

Create a screen format for the personnel database and obtain a print of your design on the printer.

(2) VIDEO STORE INVENTORY SCREEN DESIGN

Create a screen format for the video inventory database and obtain a print of your design on the printer.

(3) CHURCH CHOIR SCREEN DESIGN

Create a screen format for the church choir database and obtain a print of your design on the printer.

(4) UNDEVELOPED LEASES SCREEN DESIGN

Create a screen format for the lease database and obtain a print of your design on the printer.

UNIT TEST--Lesson 1

Name:_____ Date:_____

MULTIPLE CHOICE

Write the letter (A, B, C, or D) of the correct response to each question or statement in the blank to the left of the number (5 pts. each):

_____1. Which of the following is **not** an advantage of using a database management system?

A. Data is integrated and easier to access
B. Data is duplicated in different files
C. All applications have access to all of the data
D. Additions and changes to the data have no effect on programs

_____2. In a relational database management system, each row is a _____ and each column is a _____.

A. file, record
B. data, file
C. field, record
D. record, field

_____3. Which of the following is **not** a default file extension used by dBASE III PLUS?

A. .COM
B. .LBL
C. .PRG
D. .DBF

_____4. Which of the following is **not** true for naming fields and variables using dBASE III PLUS?

A. May be up to 10 characters long
B. May consist of both letters and numbers
C. Cannot include imbedded blanks
D. May begin with either letters or numbers

_____5. dBASE III PLUS will allow a maximum of how many database files to be open at one time?

A. 1
B. 2
C. 10
D. 64

_____6. A file which contains a set of dBASE III PLUS instructions that are stored as a program is called:

A. Command file
B. Index file
C. Database file
D. Catalog file

_____7. A file which contains the information to prepare and print a report is called:

A. Label Form file
B. Report Form file
C. Command file
D. View file

_____8. A file which contains dBASE III PLUS data in ASCII format in order to interface with other purchased software is called:

A. Database file
B. Query file
C. Text file
D. Index file

_____9. A file which stores data in records and fields is called:

A. Memory file
B. Database file
C. Index file
D. Text file

_____10. A file which provides a means to use a database in a logical order rather than the physical order is called:

A. Catalog file
B. Database file
C. Index file
D. Query file

MATCHING

Match the following field dBASE III PLUS field descriptions with the appropriate field type by writing the letter of the field type in the blank to the left of the description (5 pts. each):

_____11. Used to store the condition TRUE or FALSE

_____12. Used to store any printable character that that can be entered from the keyboard

_____13. Used to store dates

_____14. Used to store large blocks of textual information

_____15. Used to store integer or decimal numbers

a. character
b. numeric
c. date
d. logical
e. memo
f. integer
g. text

TRUE OR FALSE

Designate each of the following statements as TRUE or FALSE by writing T for True or F for False in the blank to the left of the number (5 pts. each):

_____16. dBASE III PLUS adheres to the hierarchical model for database management systems.

_____17. dBASE III PLUS has only one type of file.

_____18. File name extensions, called default extensions, are assigned by dBASE III PLUS according to the type of file being created.

_____19. If a file name extension other than the default extension is used, it must be included with the file name whenever the file is selected for use.

_____20. The structure or framework of a database file is established by defining the number records in the database.

UNIT TEST--Lesson 2

Name:_____ Date:_____

MULTIPLE CHOICE

Write the letter (A, B, C, or D) of the correct response to each question or statement in the blank to the left of the number (10 pts. each):

_____1. A vertical box underneath a command on the menu bar of the Assistant program is called:

A. Selection message
B. Pull-down menu
C. Navigation bar
D. Menu option

_____2. Two ways to give a dBASE command is to build a command using the Assistant or:

A. Use numbers in place of commands
B. Use a word processing program to write your commands
C. Enter commands interactively at the dot prompt
D. Make up your own commands using any English words

_____3. The dot prompt:

A. Indicates that dBASE III PLUS is ready to accept a command
B. Indicates that dBASE III PLUS is through processing
C. Indicates that an error has been made
D. is displayed when the Assistant program is running

_____4. If you wanted an explanation of a dBASE command called APPEND, you could obtain on-screen help for that command by typing:

A. ASSIST
B. ASSIST APPEND
C. HELP APPEND
D. HELP ASSIST

_____5. Which dBASE command should be entered each time you start the dBASE III PLUS program?

A. SET PRINT ON
B. SET DEFAULT OFF
C. SET MENU TO B:
D. SET DEFAULT TO B: (or A:)

UNIT TEST--Lesson 2 Continued

MATCHING

Identify the following selection bar movement functions by writing the letter of the key name on the line to the left of the movement description (5 pts. each):

_____ 6. Selects the currently highlighted option

_____ 7. Move the selection bar up a pull-down menu

_____ 8. Move to the next pull-down menu on the right

_____ 9. Display information relevant to the current screen

_____10. Move to the next pull-down menu on the left

a. <Up>
b. <Down>
c. <Right>
d. <Left>
e. <F1>
f. <Return>

TRUE OR FALSE

Designate each of the following statements as TRUE or FALSE by writing T for True or F for False in the blank to the left of the number (5 pts. each):

_____11. The Assistant program will help you build dBASE III PLUS commands.

_____12. Two ways to give a dBASE III PLUS command are with the aid of the Assistant program and with the aid of the HELP screen.

_____13. Interactive command entry is performed by typing dBASE III PLUS commands at the dot prompt.

_____14. Whenever starting dBASE III PLUS, your first command should be to set the default drive.

_____15. Once you leave the Assistant program, it is not possible to re-activate it without powering off the computer and re-starting dBASE III PLUS.

UNIT TEST--Lesson 3

Name:_____ Date:_____

SHORT ANSWER

Complete each of the following by writing the complete dBASE III PLUS command to initiate each of the operations (10 pts. each):

1. Create a database file named, MAILLIST:_____

_____

2. Select the database file, RENTALS, as the active database:_____

_____

3. Display on the screen the structure of the active database file:

_____

4. List on the printer the structure of the active database file:

_____

5. Change the structure of the active database file:_____

_____

6. End dBASE III processing:_____

TRUE OR FALSE

Designate each of the following statements as TRUE or FALSE by writing T for True or F for False in the blank to the left of the number (10 pts. each):

_____ 7. To create a database, you first define its structure then enter the data.

_____ 8. To define a database structure, you tell dBASE III PLUS the field name, type, width, and number of decimals for each field.

_____ 9. Each numeric field must have a least one decimal place.

_____10. Whenever you perform any operation to the data in a database file, you must first open that file for use.

UNIT TEST--Lesson 4

Name:_____ Date:_____

SHORT ANSWER

Assume that the database file, INVENT, is the active database and that it consists of the following fields:

Field Name	Type	Width	Decimals
PART_NO	C	6	
DESC	C	20	
COST	N	8	3
PRICE	N	8	2
ON_HAND	N	5	0
MINIMUM	N	5	0

Complete each of the following by writing the complete dBASE III PLUS command to initiate each of the operations (10 pts. each):

1. Add new records to the database:_____

2. Display all the records in the database on the screen:_____

   _____

3. Display all the records in the database on the screen without the record number:_____

4. List all the records in the database on the printer:_____

   _____

5. Move the record pointer to the bottom of the database and display the record (two commands):_____

   _____

6. Set the record pointer to record 10 and display the record (two commands:_____

7. Move the record pointer to the next record no matter where the pointer is currently positioned and display the record (two commands):_____

   _____

8. Display all records where the ON_HAND is less than or equal to the MINIMUM:_____

9. Display the PART_NO, DESC, and ON_HAND fields on the screen for all records:_____

10. Make a backup database file named, INVBAK, of the INVENT database file:_____

UNIT TEST--Lesson 5

Name:_____ Date:_____

SHORT ANSWER

Assume that the database file, INVENT, is the active database and that it consists of the following fields:

Field Name	Type	Width	Decimals
PART_NO	C	6	
DESC	C	20	
COST	N	8	3
PRICE	N	8	2
ON_HAND	N	5	0
MINIMUM	N	5	0

Complete each of the following by writing the complete dBASE III PLUS command to initiate each of the operations (10 pts. each):

1. Edit record 9 of the active database:_____

_____

2. Perform full-screen editing of the active database:_____

_____

3. Mark all records for deletion whose ON_HAND is equal to 0:_____

_____

4. "Undelete" all records marked for deletion:_____

_____

5. Mark record 35 for deletion:_____

_____

6. Remove deleted records from the active database:_____

_____

7. Which keys will terminate the editing mode and save the changes:

_____

TRUE OR FALSE

Designate each of the following statements as TRUE or FALSE by writing T for True or F for False in the blank to the left of the number (5 pts. each):

_____ 8. You must first identify the record number of the record to be modified in order to effectively use the EDIT command.

_____ 9. The DELETE command actually removes records from the database file.

_____10. There is no way to recover a "deleted" record from a database file.

_____11. Most updating commands operate only on the active record unless a scope of ALL is specified.

_____12. The PACK command may renumber the records in a database file if certain records are deleted.

_____13. It is quite simple using dBASE III PLUS to make a backup of a database file.

UNIT TEST--Lesson 6

Name:_____ Date:_____

SHORT ANSWER

Assume that the database file, INVENT, is the active database and that it consists of the following fields:

Field Name	Type	Width	Decimals
PART_NO	C	6	
DESC	C	20	
COST	N	8	3
PRICE	N	8	2
ON_HAND	N	5	0
MINIMUM	N	5	0

Complete each of the following by writing the complete dBASE III PLUS command to initiate each of the operations (10 pts. each):

1. Create report specifications for a report file named, INVRPT1:

   _____

2. Print the report file named, INVRPT1:_____

   _____

3. Print the report file named, INVRPT1, on the printer:_____

   _____

4. Print the report file named, INVRPT1, on the printer for all ON_HAND is less than or equal to MINIMUM:_____

   _____

5. Create label specifications for a label file named, INVLABEL:

   _____

6. Print the label file named, INVLABEL:_____

   _____

7. Print the label file named, INVLABEL, on the printer:_____

   _____

8. Print the label file named, INVLABEL, on the printer for all ON_HAND greater than 0:_____

TRUE OR FALSE

Designate each of the following statements as TRUE or FALSE by writing T for True of F for False in the blank to the left of the number (5 pts. each):

_____ 9. Report form specifications are stored in a form (.FRM) file.

_____10. Report form specifications can be edited directly when first creating a report, but may never be changed.

_____11. When creating report form specifications, you may only use fields from an active database.

_____12. Label specifications are stored in a form (.FRM) file.

UNIT TEST--Lesson 7

Name:_____ Date:_____

SHORT ANSWER

On the line to the right state the result of the following dBASE III PLUS mathematical expressions (6 pts. each):

(1) .? 4 + 5 * 3 _____

(2) .? 8 / 2 * 3 _____

(3) .? -4 * -2 _____

(4) .? -(4 * 2) _____

(5) .? (4 + 1)^2 _____

On the line to the right state whether the following dBASE III PLUS expressions are TRUE of FALSE (6 pts. each):

(6) .? 1 > 0 .AND. 0 > 1 _____

(7) .? 1 >= 1 .OR. 1 <> 0 _____

(8) .? 1 + 1 <= 2 * 1 _____

(9) .? 1 + 1 >= 1 .OR. 1 + 1 < 1 _____

(10) .? 2 * 3 > 5 .AND. 1 + 2 * 3 < 10 _____

MATCHING

Match the following dBASE commands with the command definition by writing the letter of the command definition that best describes the command in the blank to the left of the command (8 pts. each):

(11) _____ STORE
(12) _____ SAVE
(13) _____ RELEASE
(14) _____ RESTORE
(15) _____ REPLACE

a. Replaces contents of a field with a value
b. Store memory variables in a disk file
c. Eliminate memory variables from memory
d. Recall memory variables from a disk file
e. Place a value in a memory variable

UNIT TEST--Lesson 8

Name:_____ Date:_____

SHORT ANSWER

Assume that the database file, INVENT, is the active database and that it consists of the following fields:

Field Name	Type	Width	Decimals
PART_NO	C	6	
DESC	C	20	
COST	N	8	3
PRICE	N	8	2
ON_HAND	N	5	0
MINIMUM	N	5	0

Complete each of the following by writing the complete dBASE III PLUS command to initiate each of the operations (10 pts. each):

1. Sort on the field, DESC, to a database named, INVDESC:_____

   _____

2. Sort on the field, PRICE, in descending order to a database named INVPRICE:_____

3. Delete the database file, INVPRICE:_____

   _____

4. Create an index file named INVPART based on the field, PART_NO:

   _____

5. Select (use) the database file INV with the index file INVPART:

   _____

6. Search and position the record pointer on PART_NO, CDC-011:_____

   _____

7. Create an index file named INVPART based on the fields, DESC and PART_NO:_____

TRUE OR FALSE

Designate each of the following statements as TRUE or FALSE by writing T for True or F for False in the blank to the left of the number (5 pts. each):

_____ 8. With dBASE III PLUS the only way you can organize your database is by using the SORT command.

_____ 9. A sorted database must be searched sequentially to find data.

_____10. A drawback of sorting is that you may want to access the database on one field for one application and another field for another application which would require the database to be sorted be running each application.

_____11. A drawback of sorting is that any new records added are not in order and would require a sort every time data is added.

_____12. Indexing is a method of setting up a file using only the keys (fields) that you are interested in to order the database.

_____13. Indexing creates an auxiliary file that is used in conjunction with a database file.

UNIT TEST--Lesson 9

Name:_____ Date:_____

SHORT ANSWER

Complete each of the following by writing the complete dBASE III PLUS command to initiate each of the operations (10 pts. each):

1. Create a screen format named INVFORM:_____

_____

2. Select the format named INVFORM to use for updating the active database:_____

MATCHING

Match the following descriptions about data entry forms with the appropriate term by writing the letter of the proper data entry forms term in the blank to the left of the description (10 pts. each):

_____ 3. A blank screen where you can design your form and fit selected fields in in the form

_____ 4. A highlighted area containing symbols showing the type of character allowed in a field

_____ 5. The extension used by dBASE III PLUS for data entry screen formats

_____ 6. Any characters you type will write over what is already there

_____ 7. The field name displayed on the data entry form design screen

a. field highlight
b. overwrite
c. field label
d. blackboard
e. .SCN
f. .FMT

TRUE OR FALSE

Designate each of the following statements as TRUE or FALSE by writing T for True or F for False in the blank to the left of the number (5 pts. each):

_____ 8. You can create a data entry form for any database file.

_____ 9. When you create a data entry form, you must use all the fields in the database file in the data entry form.

_____10. Data entry forms may be used only one time for updating a database file.

_____11. X's in a field highlight indicate a numeric or date field.

_____12. The blackboard is initially in Overwrite mode.

_____13. A data entry form must contain graphics lines to work.

ANSWER KEYS FOR UNIT TESTS (01-03)

LESSON 01-DATABASE CONCEPTS

Multiple Choice	Matching	True or False
1. B	11. d	16. False
2. D	12. a	17. False
3. A	13. c	18. True
4. D	14. e	19. True
5. C	15. b	20. False
6. A		
7. B		
8. C		
9. B		
10. C		

LESSON 02-USING THE DBASE III PLUS ASSISTANT

Multiple Choice	Matching	True or False
1. B	6. f	11. True
2. C	7. a	12. False
3. A	8. c	13. True
4. C	9. e	14. True
5. D	10. d	15. False

LESSON 03-CREATING A DATABASE

Short Answer	True or False
1. CREATE MAILLIST	7. True
2. USE RENTALS	8. True
3. DISPLAY (or LIST) STRUCTURE	9. False
4. DISPLAY (or LIST) STRUCTURE TO PRINT	10. True
5. MODIFY STRUCTURE	
6. QUIT	

ANSWER KEYS FOR UNIT TESTS (04-06)

LESSON 04-ADDING DATA TO A DATABASE

Short Answer

1. APPEND
2. DISPLAY ALL (or LIST)
3. DISPLAY ALL (or LIST) OFF
4. DISPLAY ALL (or LIST) TO PRINT
5. GO (or GOTO) BOTTOM
   DISPLAY
6. GO (or GOTO) 10
   DISPLAY
7. SKIP (or GOTO NEXT)
   DISPLAY
8. DISPLAY ALL (or LIST) FOR ON_HAND<=MINIMUM
9. DISPLAY ALL (or LIST) PART_NO,DESC,ON_HAND
10. COPY TO INVBAK

LESSON 05-MODIFYING DATA IN A DATABASE

Short Answer

1. EDIT 9 (or EDIT RECORD 9)
2. BROWSE
3. DELETE (or DELETE ALL) FOR ON_HAND=0
4. RECALL ALL
5. DELETE RECORD 35
6. PACK
7. <Ctrl> <End>

True or False

8. True
9. False
10. False
11. True
12. True
13. True

LESSON 06-REPORT AND MAILING LABEL PREPARATION

Short Answer

1. CREATE REPORT INVRPT1
2. REPORT FORM INVRPT1
3. REPORT FORM INVRPT1 TO PRINT
4. REPORT FORM INVRPT1 FOR ON_HAND<=MINIMUM TO PRINT
5. CREATE LABEL INVLABEL
6. LABEL FORM INLABEL
7. LABEL FORM INVLABEL TO PRINT
8. LABEL FORM INLABEL FOR ON_HAND>0 TO PRINT

True or False

9. True
10. False
11. True
12. False

ANSWER KEYS FOR UNIT TESTS (07-09)

LESSON 07-MEMORY VARIABLES, OPERATIONS, AND FUNCTIONS

Multiple Choice	Matching
1. 19	11. e
2. 12	12. b
3. 8	13. c
4. -8	14. d
5. 25	15. a
6. FALSE	
7. TRUE	
8. TRUE	
9. TRUE	
10. TRUE	

LESSON 08-SORTING AND INDEXING DATABASE FILES

Short Answer	True or False
1. SORT ON DESC TO INVDESC	8. False
2. SORT ON PRICE TO INVPRICE DESCENDING	9. True
3. DELETE FILE INVPRICE.DBF	10. True
4. INDEX ON PART_NO TO INVPART	11. True
5. USE INV INDEX INVPART	12. True
6. SEEK "CDC-011"	13. True
7. INDEX ON DESC+PART_NO TO INVPART	

LESSON 09-DATA ENTRY FORMS

Short Answer	Matching	True or False
1. CREATE SCREEN INVFORM	3. d	8. True
2. SET FORMAT TO INVFORM	4. a	9. False
	5. f	10. False
	6. b	11. False
	7. c	12. True
		13. False

ANSWER KEYS FOR EXERCISES (01-03)

LESSON 01-DATABASE CONCEPTS

(1) EMPLOYEE

RECNO	SSNO	NAME	HIRED	JOB	DEPT	SALARY
1	445-24-6789	Bill Bagley	05/18/86	Salesman	Marketing	22000
2	588-89-9223	Carol Carter	04/22/86	Engineer	Research	25000
3	889-34-0182	Dave Defore	08/13/85	Clerk	Accounting	18000

(2)
a. Database File Name
b. Field Name
c. Record
d. Field

LESSON 02-USING THE DBASE III PLUS ASSISTANT

a. Menu Bar
b. Pull-Down Menu
c. Command Line
d. Status Bar
e. Navigation Line
f. Selection Message

LESSON 03-CREATING A DATABASE

EXERCISE 1:

Field Name	Type	Width	Decimals	Purpose/Function
EMPLOYEE	Character	20		Employee First/Last Name
TITLE	Character	10		Job Title
DEPT	Character	7		Department
LEVEL	Character	1		Job Level
SEX	Character	1		Sex (M or F)
YEARS	Numeric	2	0	Years Employed
SALARY	Numeric	2	8	Annual Salary

EXERCISE 2:

Field Name	Type	Width	Decimals	Purpose/Function
TAPE	Numeric	4	0	Tape Number
TITLE	Character	20		Movie Title
FORMAT	Character	4		Tape Format
RATING	Character	4		Movie Rating
CATEGORY	Character	8		Movie Classification
SHELF	Numeric	3	0	Shelf Location
PRICE	Numeric	5	2	List Price
COST	Numeric	5	2	Unit Cost

Student answers may vary. A print of the structure should be included.

FILE STRUCTURE PLANNING CHART

PLANNER'S NAME: _____ DATE: _____

DATABASE NAME: _____ DESCRIPTION: _____

_____

INDEX FILES: _____ REPORT FILES: _____

_____ _____

_____ _____

FIELD	FIELD NAME	TYPE	WIDTH	DECIMALS	PURPOSE/FUNCTION
1					
2					
3					
4					
5					
6					
7					
8					
9					
10					
11					
12					
13					
14					
15					
16					